

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

Seth W. Wiener (SBN 203747)
seth@sethwienerlaw.com
LAW OFFICES OF SETH WIENER
609 Karina Court
San Ramon, CA 94582
Telephone: (925) 487-5607

Jeffrey Francis Craft (SBN 147186)
jcraft@devlinlawfirm.com
DEVLIN LAW FIRM LLC
1731 Fox Springs Circle,
Newbury Park, CA 91320

Timothy Devlin (*pro hac vice* to be filed)
tdevlin@devlinlawfirm.com
Robert Kiddie (*pro hac vice* to be filed)
rkiddie@devlinlawfirm.com
DEVLIN LAW FIRM LLC
1526 Gilpin Avenue
Wilmington, DE 19806
Telephone: (302) 449-9010
Facsimile: (302) 353-4251

Attorneys for Express Mobile, Inc.

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
_____ DIVISION

EXPRESS MOBILE, INC.,
Plaintiff,

vs.

RISHABH BUSINESS SOLUTIONS, INC.
Defendant.

) Case No.: 3:19-cv-03356
)
)
)
) **COMPLAINT FOR PATENT**
) **INFRINGEMENT**
)
) DEMAND FOR JURY TRIAL
)
)
)
)

1 Plaintiff Express Mobile, Inc. (“Express Mobile” or “Plaintiff”), for its Complaint against
2 Defendant Rishabh Business Solutions, Inc. (“Rishabh” or “Defendant”) alleges the following:

3 **NATURE OF THE ACTION**

4 1. This is an action for patent infringement arising under the Patent Laws of the United
5 States, 35 U.S.C. § 1 *et seq.*

6 **THE PARTIES**

7 2. Plaintiff is a corporation organized under the laws of the State of Delaware with a
8 place of business at 3415 Custer Rd. Suite 104, Plano, TX 75023.

9 3. Upon information and belief, Rishabh is a corporation organized and existing under
10 the laws of California, with a place of business at 2690 S. White Rd., Ste. 245, San Jose, CA 95148
11 and can be served through its registered agent, Samil Shah, 2690 S. White Rd., Ste. 245, San Jose,
12 CA 95148.

13 4. Upon information and belief, Rishabh sells and offers to sell products and services
14 throughout the United States, including in this judicial district, and introduces products and services
15 into the stream of commerce and that incorporate infringing technology knowing that they would be
16 sold in this judicial district and elsewhere in the United States.

17 **JURISDICTION AND VENUE**

18 5. This is an action for patent infringement arising under the Patent Laws of the United
19 States, Title 35 of the United States Code.

20 6. This Court has subject matter jurisdiction under 28 U.S.C. §§ 1331 and 1338(a).

21 7. Venue is proper in this judicial district under 28 U.S.C. §1400(b). On information
22 and belief, Defendant is incorporated in California, has committed acts of infringement in this
23 District, and has a regular and established place of business within this District.

24 8. On information and belief, Defendant is subject to this Court’s general and specific
25 personal jurisdiction because Defendant has sufficient minimum contacts within the State of
26 California and this District, pursuant to due process and/or the California Long Arm Statute because
27 Defendant purposefully availed itself of the privileges of conducting business in the State of
28 California and in this District, because Defendant regularly conducts and solicits business within the

1 State of California and within this District, and because Plaintiff's causes of action arise directly
2 from each of Defendant's business contacts and other activities in the State of California and this
3 District.

4 **COUNT I – INFRINGEMENT OF U.S. Patent No. 6,546,397**

5 9. The allegations set forth in the foregoing paragraphs 1 through 8 are incorporated into
6 this First Claim for Relief.

7 10. On April 8, 2003, U.S. Patent No. 6,546,397 ("the '397 patent"), entitled "*Browser*
8 *Based Web Site Generation Tool and Run Time Engine*," was duly and legally issued by the United
9 States Patent and Trademark Office. A true and correct copy of the '397 patent is attached as
10 Exhibit A.

11 11. The inventions of the '397 patent resolve technical problems related to website
12 creation and generation. For example, the inventions enable the creation of websites through
13 browser-based visual editing tools such as selectable settings panels which describe website
14 elements, with one or more settings corresponding to commands, which features are exclusively
15 implemented utilizing computer technology including a virtual machine.

16 12. The claims of the '397 patent do not merely recite the performance of some business
17 practice known from the pre-Internet world along with the requirement to perform it on the Internet.
18 Instead, the claims of the '397 patent recite one or more inventive concepts that are rooted in
19 computerized website creation technology, and overcome problems specifically arising in the realm
20 of computerized website creation technologies.

21 13. The claims of the '397 patent recite an invention that is not merely the routine or
22 conventional use of website creation systems and methods. Instead, the invention describes a
23 browser-based website creation system and method in which the user-selected settings representing
24 website elements are stored in a database, and in which said stored information is retrieved to
25 generate said website.

26 14. The technology claimed in the '397 patent does not preempt all ways of using website
27 or web page authoring tools nor preempt the use of all website or web page authoring tools, nor
28 preempt any other well-known or prior art technology.

1 15. Accordingly, each claim of the '397 patent recites a combination of elements
2 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an
3 ineligible concept.

4 16. In C.A. 2:17-00128, a case filed in the Eastern District of Texas, the defendant in that
5 action, KTree Computer Solutions brought a Motion for Judgment on the Pleadings asserting that the
6 '397 patent, along with U.S. Patent No. 7,594,168 (asserted in Count II below) were invalid as
7 claiming abstract subject matter under 35 U.S.C. § 101. (C.A. 2:17-00128 Dkt. 9.) Subsequent
8 briefing included Plaintiff's Response and related Declarations and Exhibits (C.A. 2:17-00128 Dkt.
9 17, 22-24), KTree's Reply (C.A. 2:17-00128 Dkt. 25), and Plaintiff's Sur-Reply and related
10 Declarations and Exhibits (C.A. 2:17-00128 Dkt. 26-27). Each of those filings is incorporated by
11 reference into this Complaint.

12 17. After a consideration of the respective pleadings, Magistrate Judge Payne
13 recommended denial of KTree's motion, without prejudice, holding that "the claims appear to
14 address a problem particular to the internet: dynamically generating websites and displaying web
15 pages based on stored user-selected settings" and further stating "the asserted claims do not bear all
16 of the hallmarks of claims that have been invalidated on the pleadings by other courts in the past.
17 For example, the claims are not merely do-it-on-a-computer claims." (C.A. 2:17-00128 Dkt. 29
18 attached hereto as Exhibit B.) Judge Payne's report and recommendation is incorporated by
19 reference into this Complaint. No objection was filed to the Magistrate Judge's report and
20 recommendation and the decision therefore became final.

21 18. In C.A. Nos. 3:18-cv-04679 and 3:18-04688, both of which were filed in the Northern
22 District of California, the respective defendant in each of those actions brought a Motion to Dismiss
23 asserting that the '397 patent, along with U.S. Patent No. 7,594,168 (asserted in Count II below)
24 were invalid as claiming abstract subject matter under 35 U.S.C. § 101. The § 101 briefing in each
25 of those cases is incorporated by reference into this Complaint.

26 19. After consideration of the respective pleadings and oral argument, Judge Richard
27 Seeborg issued orders denying each respective motion to dismiss drawing a comparison between the
28 asserted Express Mobile patents with those patents asserted in *Enfish, LLC v. Microsoft Corp.*, 822

1 F.3d 1327 (Fed. Cir. 2016). (C.A. 3:18 -04679 Dkt. 45, attached hereto as Exhibit C, and C.A. 3:18-
2 04688 Dkt. 40, attached hereto as Exhibit D.)

3 20. Plaintiff is the assignee and owner of the right, title and interest in and to the '397
4 patent, including the right to assert all causes of action arising under said patents and the right to any
5 remedies for infringement of them.

6 21. Upon information and belief, Defendant has and continues to directly infringe at least
7 claims 1-6, 8-11, 14-15, 17, 20, 24-25, 35, and 37 of the '397 patent by using a browser-based
8 website and/or web page authoring tool in which the user-selected settings representing website
9 elements are stored in a database, and in which said stored information is retrieved to generate said
10 website (the "Accused Instrumentalities"). The Accused Instrumentalities include but are not
11 limited to the website building tools used and/or provided by Defendant, such as, for example
12 Drupal, Magento, Shopify and/or Wordpress. *See, e.g.*, [https://www.rishabhsoft.com/web-](https://www.rishabhsoft.com/web-application/drupal-development-services)
13 [application/drupal-development-services](https://www.rishabhsoft.com/magento-development-services); [https://www.rishabhsoft.com/magento-development-](https://www.rishabhsoft.com/magento-development-services)
14 [services](https://www.rishabhsoft.com/web-application/ecommerce-development); <https://www.rishabhsoft.com/web-application/ecommerce-development>;
15 <https://www.rishabhsoft.com/web-application/portals-collaboration>.

16 22. On information and belief, Defendant is a for-profit organization with revenues of
17 approximately \$75 million U.S.D. per year. Moreover, Defendant, its employees and/or agents
18 utilize the Accused Instrumentalities in the building and/or hosting of websites for Defendant's
19 customers, leading to direct or indirect revenues and profit. As one example of indirect profit,
20 entities such as Defendant will frequently offer website building and/or hosting services at reduced
21 pricing as an inducement to attract customers, who then purchase additional products or services.
22 On information and belief, without the availability of infringing tools such as the Accused
23 Instrumentalities, Defendant would be at a disadvantage in the marketplace and would generate less
24 revenue overall.

25 23. In particular, claim 1 of the '397 patent generally recites a method enabling
26 production of websites on and for computers with browsers and virtual machines, by presenting,
27 through a browser, a selectable settings menu describing elements, such setting(s) corresponding to
28 commands to the virtual machine; generating a display in accordance with selected settings; storing

1 information regarding selected settings in a database; generating a website at least in part by
2 retrieving said information; and building web page(s) to generate said website and a run time file,
3 where the run time file uses the stored information to generate virtual machine commands for the
4 display of at least a portion of web page(s).

5 24. The Accused Instrumentalities infringe claim 1 of the '397 patent through a
6 combination of features which collectively practice each limitation of claim 1. By way of example,
7 modern internet browsers such as Microsoft Internet Explorer, Mozilla's Firefox, Apple Safari,
8 Google Chrome, and Opera include virtual machines within the meaning of the '397 patent. (*See*,
9 *e.g.*, <http://developer.telerik.com/featured/a-guide-to-javascript-engines-for-idiots/>;
10 <http://dictionary.reference.com/browse/virtual+machine?s=t>). The Accused Instrumentalities
11 support the use of the latest versions of Internet Explorer 11 or later, Microsoft Edge, latest-1,
12 Firefox latest, latest-1, Chrome latest, latest-1, Safari latest, latest-1 (Mac OS), Safari Mobile for
13 iPad 2, iPad Mini, iPad with Retina Display (iOS 7 or later), for desktop site, Safari Mobile for
14 iPhone 4 or later; iOS 7 or later, for mobile site, Chrome for mobile latest-1 (Android 4 or later) for
15 mobile site, where *latest-1* means one major version earlier than the latest released version. (*See*,
16 *e.g.*, <https://www.drupal.org/docs/8/system-requirements/browser->;
17 http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html;
18 <http://theforest.net/category/wordpress>; [http://devdocs.magento.com/guides/v2.0/install-](http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html)
19 [gde/system-requirements_browsers.html](http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html).) All of these browsers rely on browser engines comprising
20 virtual machines to interpret and execute JavaScript and HTML to render web pages on a computer.

21 25. By way of further example, the Accused Instrumentalities enable users to produce
22 websites through browsers on users' computers via interaction with an Internet server. For example,
23 in order to add a new page to a user's website, the user logs in and then a server of the Accused
24 Instrumentalities initiates presentation to the user through a browser of a website-builder tool. From
25 the interface—sometimes referred to as a dashboard—of the Accused Instrumentalities, the user can
26 navigate and add elements and element properties commensurate with a new page. A display is
27 generated in accordance with one or more user selected settings substantially contemporaneously
28 with the selection thereof. This is performed, for example, using a visual editing tool through a

1 browser. The WYSIWYG interface for selecting center alignment of an image can also be accessed,
2 and then the user can select various options such as a font and paragraph styles. After the user
3 selects options such as image/text alignment or font and paragraph styles through the WYSIWYG
4 editor, the display immediately updates to reflect the selected option. Furthermore, when images are
5 uploaded by a user, those images are displayed in approximately 0-2 seconds depending on file size
6 and bandwidth.

7 26. Data is stored in a database, including information corresponding to user selected
8 settings such as, for example, the selections of text color. Other user selections are also stored
9 including, for example, the layout, image filenames, thumbnails, and paragraph margin settings for
10 defining the alignment of an image location. The Accused Instrumentalities build one or more web
11 pages to generate a website from at least a portion of a database and at least one run time file, where
12 at least one run time file utilizes information stored in said database to generate virtual machine
13 commands for the display of at least a portion of said one or more web pages.

14 27. At run time, at least some of these files use information stored in the database to
15 generate the HTML for the final rendered HTML page. This HTML represents virtual machine
16 commands for display of the page because it is read and used by the applicable browser's engine,
17 including a virtual machine, in order to render the page. On information and belief, the Accused
18 Instrumentalities further rely on the browser engine's component JavaScript engine to either display
19 a portion of the page directly, or generate HTML to be executed for display by the main layout
20 engine.

21 28. Additionally, the "PHP code," including the PHP template files, can be viewed in the
22 file directory for the Accused Instrumentalities, and this directory includes various other runtime
23 files (including other PHP files, JavaScript files, PHTML, and/or XML). It follows that a user will
24 view the finalized website developed with said tools in a browser outside of the website authoring
25 environment to verify the website conforms to the intended design. *See, e.g.,*
26 <https://techterms.com/definition/runtime>.

27 29. The presence of the above referenced elements are demonstrated, by way of example,
28 by reference to publicly available information. Regarding Drupal, *see, e.g.,*

1 <https://www.drupal.org/home>; [https://www.drupal.org/docs/8/system-requirements/browser-](https://www.drupal.org/docs/8/system-requirements/browser-requirements)
2 [requirements; https://www.drupal.org/project/ckeditor](https://www.drupal.org/project/ckeditor);
3 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
4 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
5 [images/10/03/2016/9821](#); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
6 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
7 <https://www.drupal.org/docs/8/system-requirements/web-server>;
8 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
9 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
10 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
11 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding
12 Magento, *see, e.g.*, [http://docs.magento.com/m1/ee/user_guide/system-operations/browser-](http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html)
13 [capabilities-detection.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html); [http://docs.magento.com/m1/ee/user_guide/system-operations/index-](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html)
14 [management.html](http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html); http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html;
15 http://docs.magento.com/m1/ce/user_guide/cms/page-create.html;
16 http://docs.magento.com/m1/ce/user_guide/cms/editor.html;
17 http://docs.magento.com/m1/ce/user_guide/cms/links.html;
18 http://docs.magento.com/m1/ce/user_guide/cms/editor-insert-image.html;
19 http://docs.magento.com/m1/ce/user_guide/cms/editor-add-widget.html;
20 http://docs.magento.com/m1/ce/user_guide/design/page-layout.html;
21 http://docs.magento.com/m1/ce/user_guide/design/layout-updates.html;
22 http://docs.magento.com/m1/ee/user_guide/store-operations/stores-multiple.html;
23 http://docs.magento.com/m1/ee/user_guide/store-operations/store-hierarchy.html;
24 http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html. Regarding
25 Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;
26 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Hierarchy;
27 http://codex.wordpress.org/Function_Reference/the_title;
28 http://codex.wordpress.org/Function_Reference/the_content;

1 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
2 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
3 http://codex.wordpress.org/Query_Overview.

4 30. Claim 2 of the '397 patent generally recites an apparatus for producing websites on
5 and for computers having a browser and a virtual machine, said apparatus comprising an interface to
6 present a settings menu which describes elements, said panel presented through a browser, where the
7 selectable setting(s) corresponds to commands to the virtual machine; a browser to generate a
8 display in accordance with selected setting(s); a database for storing information regarding selected
9 settings; and a build tool having run time file(s) for generating web page(s) and using stored
10 information to generate commands to the virtual machine for generating at least a portion of web
11 page(s).

12 31. The Accused Instrumentalities infringe claim 2 of the '397 patent through a
13 combination of features which collectively practice each limitation of claim 2. By way of example,
14 modern internet browsers such as Microsoft Internet Explorer, Mozilla's Firefox, Apple Safari,
15 Google Chrome, and Opera include virtual machines within the meaning of the '397 patent. (*See*,
16 *e.g.*, <http://developer.telerik.com/featured/a-guide-to-javascript-engines-for-idiots/>;
17 <http://dictionary.reference.com/browse/virtual+machine?s=t>). The Accused Instrumentalities
18 support the use of the latest versions of Internet Explorer 11 or later, Microsoft Edge, latest-1,
19 Firefox latest, latest-1, Chrome latest, latest-1, Safari latest, latest-1 (Mac OS), Safari Mobile for
20 iPad 2, iPad Mini, iPad with Retina Display (iOS 7 or later), for desktop site, Safari Mobile for
21 iPhone 4 or later; iOS 7 or later, for mobile site, Chrome for mobile latest-1 (Android 4 or later) for
22 mobile site, where *latest-1* means one major version earlier than the latest released version. (*See*,
23 *e.g.*, <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
24 http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html;
25 <http://themeforest.net/category/wordpress>; http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html.) All of these browsers rely on browser engines comprising
26 virtual machines to interpret and execute JavaScript and HTML to render web pages on a computer.
27
28

1 32. By way of further example, the Accused Instrumentalities enable users to produce
2 websites through browsers on users' computers via interaction with an Internet server. For example,
3 in order to add a new page to a user's website, the user logs in and then a server of the Accused
4 Instrumentalities initiates presentation to the user through a browser of a website-builder tool. From
5 the interface—sometimes referred to as a dashboard—of the Accused Instrumentalities, the user can
6 navigate and add elements and element properties commensurate with a new page. A display is
7 generated in accordance with one or more user selected settings substantially contemporaneously
8 with the selection thereof. This is performed, for example, using a visual editing tool through a
9 browser. The WYSIWYG interface for selecting center alignment of an image can also be accessed,
10 and then the user can select various options such as a font and paragraph styles. After the user
11 selects options such as image/text alignment or font and paragraph styles through the WYSIWYG
12 editor, the display immediately updates to reflect the selected option. Furthermore, when images are
13 uploaded by a user, those images are displayed in approximately 0-2 seconds depending on file size
14 and bandwidth.

15 33. Data is stored in a database, including information corresponding to user selected
16 settings such as, for example, the selections of text color. Other user selections are also stored
17 including, for example, the layout, image filenames, thumbnails, and paragraph margin settings for
18 defining the alignment of an image location. The Accused Instrumentalities build one or more web
19 pages to generate a website from at least a portion of a database and at least one run time file, where
20 at least one run time file utilizes information stored in said database to generate virtual machine
21 commands for the display of at least a portion of said one or more web pages.

22 34. At run time, at least some of these files use information stored in the database to
23 generate the HTML for the final rendered HTML page. This HTML represents virtual machine
24 commands for display of the page because it is read and used by the applicable browser's engine,
25 including a virtual machine, in order to render the page. On information and belief, the Accused
26 Instrumentalities further rely on the browser engine's component JavaScript engine to either display
27 a portion of the page directly, or generate HTML to be executed for display by the main layout
28 engine.

1 35. Additionally, the “PHP code,” including the PHP template files, can be viewed in the
2 file directory for the Accused Instrumentalities, and this directory includes various other runtime
3 files (including other PHP files, JavaScript files, PHTML, and XML). *See, e.g.*,
4 <https://techterms.com/definition/runtime>.

5 36. It follows that a user will view the finalized website developed with said tools in a
6 browser outside of the website authoring environment to verify the website conforms to the intended
7 design.

8 37. The presence of the above referenced elements are demonstrated, by way of example,
9 by reference to publicly available information. Regarding Drupal, *see, e.g.*,

10 <https://www.drupal.org/home>; [https://www.drupal.org/docs/8/system-requirements/browser-](https://www.drupal.org/docs/8/system-requirements/browser-requirements)
11 [requirements](https://www.drupal.org/project/ckeditor); <https://www.drupal.org/project/ckeditor>;

12 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;

13 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
14 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);

15 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;

16 <https://www.drupal.org/docs/8/system-requirements/web-server>;

17 <https://www.drupal.org/docs/8/core/modules/rest/overview>;

18 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;

19 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;

20 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding

21 Magento, *see, e.g.*, [http://docs.magento.com/m1/ee/user_guide/system-operations/browser-](http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html)
22 [capabilities-detection.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html); [http://docs.magento.com/m1/ee/user_guide/system-operations/index-](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html)

23 [management.html](http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html); http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html;

24 http://docs.magento.com/m1/ce/user_guide/cms/page-create.html;

25 http://docs.magento.com/m1/ce/user_guide/cms/editor.html;

26 http://docs.magento.com/m1/ce/user_guide/cms/links.html;

27 http://docs.magento.com/m1/ce/user_guide/cms/editor-insert-image.html;

28 http://docs.magento.com/m1/ce/user_guide/cms/editor-add-widget.html;

1 http://docs.magento.com/m1/ce/user_guide/design/page-layout.html;
2 http://docs.magento.com/m1/ce/user_guide/design/layout-updates.html;
3 http://docs.magento.com/m1/ee/user_guide/store-operations/stores-multiple.html;
4 http://docs.magento.com/m1/ee/user_guide/store-operations/store-hierarchy.html;
5 http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html. Regarding
6 Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;
7 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Hierarchy;
8 http://codex.wordpress.org/Function_Reference/the_title;
9 http://codex.wordpress.org/Function_Reference/the_content;
10 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
11 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
12 http://codex.wordpress.org/Query_Overview.

13 38. Claim 3 of the '397 patent recites the apparatus of claim 2, wherein the database is a
14 multi-dimensional array structured database.

15 39. The Accused Instrumentalities infringe claim 3 of the '397 patent through, by way of
16 example, patent through a combination of features which collectively practice each limitation of
17 claim 3.

18 40. By way of example, the JSON strings that are used to generate, in part, field
19 capabilities originate from the database and therefore reflect the database structure and contents
20 showing, on information and belief, the implementation of a multidimensional array structured
21 database. By way of further evidence, the JSON strings show that there are dimensions for various
22 parameters. *See, e.g.*, <https://www.drupal.org/files/issues/Field.png>;
23 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>;
24 <http://devdocs.magento.com/guides/v2.0/get-started/gs-web-api-request.html>;
25 <https://code.tutsplus.com/>; [https://wordpress.stackexchange.com/questions/43302/wordpress-](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)
26 [settings-api-and-option-array-structure](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure).

1 41. Claim 4 of the '397 patent recites the apparatus of claim 3, wherein the representative
2 information is Boolean data, numeric data, string data or multi-dimensional arrays of various
3 multimedia objects.

4 42. The Accused Instrumentalities infringe claim 4 of the '397 patent through a
5 combination of features that practice the limitations of Claim 4. *See, e.g.*,
6 <https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions>;
7 <http://devdocs.magento.com/guides/m1x/api/rest/Resources/Products/products.html>;
8 [https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)
9 [structure.](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)

10 43. Claim 5 of the '397 patent recites the apparatus of claim 4, wherein said elements
11 include multimedia objects selected from the group consisting of a color, a font, an image, an audio
12 clip, a video clip, a text area and a URL.

13 44. The Accused Instrumentalities infringe claim 5 of the '397 patent through a
14 combination of features that practice the limitations of Claim 5.

15 45. By way of example, the Accused Instrumentalities include various multimedia objects
16 selected from a group contained within a WYSIWYG Editor. Examples include color, font, an
17 image, a video, a text area and a URL as they appear in the WYSIWYG Editor. The multimedia
18 objects created in the WYSIWYG editor are stored in the database and appear as HTML scripted
19 text in the database. Text and vector objects can be selected and colored by selecting them or “click
20 and dragging” over them in the WYSIWYG editor. A color may also be selected from the color
21 dropdowns on the control bar of the Editor. This color is saved to the database; as part of the HTML
22 of the description record. Moreover, text objects may be assigned a font by making such a selection
23 or “click and dragging” over them in the WYSIWYG editor. A font can then be selected from the
24 font dropdown on the control bar of the Editor. This font selection is thereafter saved to the database
25 as part of the HTML of the description record. Selecting the Image button in the WYSIWYG editor
26 opens a tabbed panel where the user designates source, title, format, size, etc. The image file is
27 uploaded to the server and the file’s location and style are saved and posted to the database as part of
28 the HTML of the description record. Furthermore, videos are created by clicking on the Media

1 module, which opens a tabbed panel where the user designates URL, format, size, etc. The video's
2 URL and style elements are saved to the database as part of the HTML of the description record. A
3 text area may also be selected for creation by clicking in the frame of the WYSIWYG Editor and
4 typing. The text and its style are saved to the database as part of the HTML of the description
5 record. After entering text into the WYSIWYG editor's text area, a URL assigned by clicking and
6 dragging over the text object you wish to link, and then selecting the "chain" link button from the
7 control bar; which opens a tabbed panel where the user can designate the URL, target, etc. The text
8 and its style are saved to the database as part of the HTML of the description record.

9 46. The presence of the above referenced elements are demonstrated, by way of example,
10 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/project/ckeditor>;
11 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
12 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
13 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
14 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
15 [images/10/03/2016/9821](https://www.drupal.org/docs/8/core/modules/media/overview); <https://www.drupal.org/docs/8/core/modules/media/overview>;
16 https://www.drupal.org/project/media_entity.

17 47. Claim 6 of the '397 patent recites the apparatus of claim 2, wherein said elements are
18 selected from the group consisting of a button, an image, a paragraph, a frame, a table, a form and a
19 vector object.

20 48. The Accused Instrumentalities infringe claim 6 of the '397 patent through a
21 combination of features that practice the limitations of Claim 6.

22 49. By way of example, the Accused Instrumentalities include various user selectable
23 menus where various elements can be placed on a web page. Those various user selectable menus
24 are used to place elements selected from the group consisting of a button, an image, a paragraph, a
25 frame, a table, a form and a vector object. The cells of a table and maps would reside in a frame, and
26 that, dividers, maps and the lines in tables would be, at least in part, vector objects.

27 50. The presence of the above referenced elements are demonstrated, by way of example,
28 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/project/ckeditor>;

1 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
2 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
3 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
4 https://www.drupal.org/docs/8/core/modules/custom_block/overview;
5 <https://www.drupal.org/docs/8/core/modules/contact/overview>.

6 51. Claim 8 recites the apparatus of claim 2, wherein said elements include one or more
7 objects on a web page, and wherein said description of elements are a transition or an animation of at
8 least one of said elements on a web page.

9 52. The Accused Instrumentalities infringe claim 8 of the '397 patent through a
10 combination of features which collectively practice each limitation of claim 8. *See, e.g.*,
11 <https://wordpress.org/plugins/animate-everything/>.

12 53. Claim 9 recites the apparatus of claim 2, wherein said elements include a button or an
13 images, wherein said selectable settings include the selection of an element style, and wherein said
14 build tool includes means for storing information representative of selected style in a database.

15 54. The Accused Instrumentalities infringe claim 9 of the '397 patent through a
16 combination of features which collectively practice each limitation of claim 9. *See, e.g.*,
17 <https://www.drupal.org/project/ckeditor>;
18 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
19 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
20 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); <https://www.drupal.org/docs/8/core/modules/image/working-with-images>;
21 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
22 <https://www.drupal.org/docs/8/system-requirements/web-server>;
23 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
24 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
25 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
26 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
27 <https://www.wpbeginner.com/wp-tutorials/how-to-add-custom-styles-to-wordpress-visual-editor/>.

28

1 55. Claim 10 recites the apparatus of claim 9, wherein said elements are described by
2 multiple object states.

3 56. The Accused Instrumentalities infringe claim 10 of the '397 patent through a
4 combination of features which collectively practice each limitation of claim 10. For example,
5 buttons can have multiple object states. *See, e.g.*, [https://www.drupal.org/docs/8/core/themes/seven-](https://www.drupal.org/docs/8/core/themes/seven-theme)
6 [theme; https://wordpress.org/plugins/animate-everything/](https://wordpress.org/plugins/animate-everything/).

7 57. Claim 11 recites the apparatus of claim 9, wherein said elements are described by a
8 transformation or a timelines of said selected styles.

9 58. The Accused Instrumentalities infringe claim 11 of the '397 patent through a
10 combination of features which collectively practice each limitation of claim 11. By way of example,
11 the Accused Instrumentalities support CSS architecture. *See, e.g.*,
12 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>; *see also, e.g.*,
13 <http://demos.dojotoolkit.org/demos/css3/demo.html>; [https://wordpress.org/plugins/animate-](https://wordpress.org/plugins/animate-everything/)
14 [everything/](https://wordpress.org/plugins/animate-everything/).

15 59. Claim 14 recites the apparatus of claim 2, wherein said elements include buttons or
16 images, wherein said description of elements is a transition or a timeline which is selected according
17 to input from a mouse, and wherein said build tool includes means for storing information
18 representative of said selected description of elements in said database.

19 60. The Accused Instrumentalities infringe claim 14 of the '397 patent through a
20 combination of features which collectively practice each limitation of claim 14.

21 61. By way of example, the Accused Instrumentalities include various CSS libraries that
22 are used extensively for adding transformations and timelines to selected elements. *See, e.g.*,
23 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
24 <http://demos.dojotoolkit.org/demos/css3/demo.html>.

25 62. Claim 15 recites the apparatus of claim 14, wherein at least one of said description of
26 elements is a timeline or an animation.

27 63. The Accused Instrumentalities infringe claim 15 of the '397 patent through a
28 combination of features which collectively practice each limitation of claim 15.

1 64. By way of example, the Accused Instrumentalities enable descriptions of elements
2 describing CSS animations. *See, e.g.*, [https://www.drupal.org/docs/develop/standards/css/css-
3 architecture-for-drupal-8](https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8); <http://demos.dojotoolkit.org/demos/css3/demo.html>.

4 65. Claim 17 recites the apparatus of claim 2, wherein one or more of said elements is a
5 button or an image, wherein said description of elements is a transition, an animation or a timeline,
6 and wherein said build engine includes means to synchronize said description of said one or more
7 elements.

8 66. The Accused Instrumentalities infringe claim 17 of the '397 patent through a
9 combination of features which collectively practice each limitation of claim 17.

10 67. By way of example, the Accused Instrumentalities enable the definition of certain
11 parent elements and child element in certain classes, which can include the URL for an image, in the
12 database, and that also can be zoomed. The parent element can include two transform styles, such as
13 one for the parent and one for its lens.

14 68. Claim 20 recites the apparatus of claim 2, wherein at least one of said elements is a
15 child button or a child object, wherein said description of said elements is a timeline, a transition or
16 an animation, and wherein said build engine includes means for defining said description of said
17 element.

18 69. The Accused Instrumentalities infringe claim 20 of the '397 patent through a
19 combination of features which collectively practice each limitation of claim 20.

20 70. By way of example, the Accused Instrumentalities enable the description of elements
21 as timelines or transition. Moreover, the build engine includes the means for defining said
22 description of said element through a choice of menu items through the design tab of the Product
23 Information Admin Panel.

24 71. Claim 24 recites the apparatus of claim 2, wherein said run time files include one
25 compressed website specific, customized run time engine program file and one compressed website
26 specific, customized run time engine library file.

27 72. The Accused Instrumentalities infringe claim 24 of the '397 patent through a
28 combination of features which collectively practice each limitation of claim 24.

1 73. By way of example, the Accused Instrumentalities include two customized runtime
2 files, an HTML file and a second unique CSS file. *See, e.g.*,
3 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
4 <https://www.drupal.org/docs/8/system-requirements/web-server>;
5 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
6 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
7 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
8 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>.

9 74. Claim 25 recites the apparatus of claim 24, wherein said run time files include a
10 dynamic web page scaling mechanism, whereby each of said one or more generated web pages is
11 scaled for viewing on said display.

12 75. The Accused Instrumentalities infringe claim 25 of the '397 patent through a
13 combination of features which collectively practice each limitation of claim 25.

14 76. By way of example, the Accused Instrumentalities enable rescaling of a web page to
15 the size of the particular screen that is being used. *See, e.g.*,
16 <https://www.drupal.org/docs/8/mobile/responsive-web-design>;
17 <https://www.drupal.org/docs/8/mobile/web-based-mobile-apps>.

18 77. Claim 35 of the '397 patent generally recites the apparatus of claim 2, wherein the
19 build tool includes dynamic resizing means operable to redefine a size of a web page upon being
20 display.

21 78. The Accused Instrumentalities infringe claim 35 of the '397 patent through a
22 combination of features which collectively practice each limitation of claim 35.

23 79. By way of example, the Accused Instrumentalities enable dynamic resizing upon
24 display to a different device and screen. For example, the Accused Instrumentalities include
25 “Responsive Web Design.” Responsive Web Design refers to web design that changes formatting
26 and lay-out to respond to different devices, screen sizes and browser capabilities. The Accused
27 Instrumentalities therefore enable the creation of web pages that may be viewed with resizing means
28 operable to redefine a size of a web page upon being displayed. *See, e.g.*,

1 http://www.w3schools.com/html/html_responsive.asp;
2 <https://www.drupal.org/docs/8/mobile/responsive-web-design>;
3 <https://www.drupal.org/docs/8/mobile/web-based-mobile-apps>.

4 80. Claim 37 of the '397 patent generally recites [a]n apparatus for producing websites
5 with web page(s) on and for a computer with a browser and a virtual machine, the apparatus
6 comprising: an interface for building a website through control of website elements, being operable
7 through the browser on to: present a selectable settings menu, accept settings, and generate the
8 display in accordance with an assembly of settings contemporaneously with the acceptance thereof,
9 at least one setting being operable to generate said display through commands to said virtual
10 machine; an internal database associated with the interface for storing information representative of
11 one or more of assembly of settings for controlling elements of the website; and a build tool to
12 construct web page(s) of the website having: an external database containing data corresponding to
13 the information stored in the internal database, and one or more run time files, where said run time
14 files use information stored in the external database to generate virtual machine commands for the
15 display of at least a portion of one or more web pages.

16 81. The Accused Instrumentalities infringe claim 37 of the '397 patent through a
17 combination of features which collectively practice each limitation of claim 37. By way of example,
18 modern internet browsers such as Microsoft Internet Explorer, Mozilla's Firefox, Apple Safari,
19 Google Chrome, and Opera include virtual machines within the meaning of the '397 patent. (*See*,
20 *e.g.*, <http://developer.telerik.com/featured/a-guide-to-javascript-engines-for-idiots/>;
21 <http://dictionary.reference.com/browse/virtual+machine?s=t>). The Accused Instrumentalities
22 support the use of the latest versions of Internet Explorer 11 or later, Microsoft Edge, latest-1,
23 Firefox latest, latest-1, Chrome latest, latest-1, Safari latest, latest-1 (Mac OS), Safari Mobile for
24 iPad 2, iPad Mini, iPad with Retina Display (iOS 7 or later), for desktop site, Safari Mobile for
25 iPhone 4 or later; iOS 7 or later, for mobile site, Chrome for mobile latest-1 (Android 4 or later) for
26 mobile site, where *latest-1* means one major version earlier than the latest released version. (*See*
27 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
28 http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html;

1 <http://theforest.net/category/wordpress>; <http://devdocs.magento.com/guides/v2.0/install->
2 [gde/system-requirements_browsers.html](http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html).)

3 82. By way of example, the Accused Instrumentalities include various multimedia objects
4 selected from a group contained within a WYSIWYG Editor. Examples include color, font, an
5 image, a video, a text area and a URL as they appear in the WYSIWYG Editor. The multimedia
6 objects created in the WYSIWYG editor are stored in the database and appear as HTML scripted
7 text in the database. Text and vector objects can be selected and colored by selecting them or
8 “clicking and dragging” over them in the WYSIWYG editor. A color may also be selected from the
9 color dropdowns on the control bar of the Editor. This color is saved to the database; as part of the
10 HTML of the description record. Moreover, text objects may be assigned a font by making such a
11 selection or “click and dragging” over them in the WYSIWYG editor. A font can then be selected
12 from the font dropdown on the control bar of the Editor. This font selection is thereafter saved to the
13 database as part of the HTML of the description record. Selecting the Image button in the
14 WYSIWYG editor opens a tabbed panel where the user designates source, title, format, size, etc.
15 The image file is uploaded to the server and the file’s location and style are saved and posted to the
16 database as part of the HTML of the description record. Furthermore, videos are created by clicking
17 on the Media module, which opens a tabbed panel where the user designates URL, format, size, etc.
18 The video’s URL and style elements are saved to the database as part of the HTML of the
19 description record. A text area may also be selected for creation by clicking in the frame of the
20 WYSIWYG Editor and typing. The text and its style are saved to the database as part of the HTML
21 of the description record. After entering text into the WYSIWYG editor’s text area, a URL assigned
22 by clicking and dragging over the text object you wish to link, and then selecting the “chain” link
23 button from the control bar; which opens a tabbed panel where the user can designate the URL,
24 target, etc. The text and its style are saved to the database as part of the HTML of the description
25 record.

26 83. Furthermore, the Accused Instrumentalities enable data from the client-side form
27 referenced to be stored in a server-side database.

28

1 84. The presence of the above referenced elements are demonstrated, by way of example,
2 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/home>;
3 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
4 <https://www.drupal.org/project/ckeditor>;
5 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
6 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
7 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
8 https://www.drupal.org/project/save_draft; [https://www.drupal.org/docs/7/understanding-](https://www.drupal.org/docs/7/understanding-drupal/technology-stack)
9 [drupal/technology-stack](https://www.drupal.org/docs/7/understanding-drupal/technology-stack); <https://www.drupal.org/docs/8/system-requirements/web-server>;
10 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
11 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
12 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
13 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
14 http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html;
15 http://docs.magento.com/m1/ee/user_guide/design/layout-updates.html;
16 [http://docs.magento.com/m1/ee/user_guide/system-](http://docs.magento.com/m1/ee/user_guide/system-operations/cache-page.html?Highlight=database%20retrieval)
17 [operations/cache-](http://docs.magento.com/m1/ee/user_guide/system-operations/cache-page.html?Highlight=database%20retrieval)
18 [page.html?Highlight=database%20retrieval](http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage.html); [operations/media-storage.html](http://docs.magento.com/m1/ee/user_guide/system-
19 <a href=); [operations/media-storage-database.html](http://docs.magento.com/m1/ee/user_guide/system-
20 <a href=);
21 [http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.p](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf)
22 [df](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf); http://docs.magento.com/m1/ee/user_guide/store-operations/stores-multiple.html;
23 http://docs.magento.com/m1/ee/user_guide/store-operations/store-hierarchy.html;
24 http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html;
25 http://docs.magento.com/m1/ce/user_guide/design/page-layout.html;
26 http://docs.magento.com/m1/ce/user_guide/design/layout-updates.html; and
27 [http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.p](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf)
28 [df](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf). Regarding Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;
<http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Hierarchy;

1 http://codex.wordpress.org/Function_Reference/the_title;
2 http://codex.wordpress.org/Function_Reference/the_content;
3 http://codex.wordpress.org/Template_Tags/get_the_title;
4 http://codex.wordpress.org/Query_Overview; <https://www.wpbeginner.com/glossary/database/>; and
5 <https://codex.wordpress.org/Pages>.

6 85. Upon information and belief, these Accused Instrumentalities are used, marketed,
7 provided to, and or used by or for each of Defendant's partners, clients, customers, and/or end users
8 across the country and in this District.

9 86. In particular, Defendant's actions that aid and abet others such as its partners,
10 customers, clients, and/or end users to infringe include advertising and distributing the Accused
11 Instrumentalities and providing instruction materials, training, and services regarding the Accused
12 Instrumentalities. *See, e.g.*, <https://www.rishabhsoft.com/blog/wordpress-or-drupal>. On information
13 and belief, Defendant has engaged in such actions with specific intent to cause infringement or with
14 willful blindness to the resulting infringement because Defendant has had actual knowledge of the
15 '397 patent and knowledge that its acts were inducing infringement of the '397 patent since at least
16 the date Rishabh received notice that such activities infringed the '397 patent.

17 87. Upon information and belief, Defendant is liable as a contributory infringer of the
18 '397 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
19 website authoring tools to be especially made or adapted for use in an infringement of the '397
20 patent. The Accused Instrumentalities are a material component for use in practicing the '397 patent
21 and are specifically made and are not a staple article of commerce suitable for substantial non-
22 infringing use.

23 88. Defendant was made aware of the '397 patent and its infringement thereof at least as
24 early as the filing of this Complaint.

25 89. Since the date of the filing of this Complaint, Defendant's infringement of the '397
26 patent has been willful.

27
28

1 90. Plaintiff has not sold any product nor offered a service within the scope of any claim
2 of the '397 patent. In addition, prior to August 12, 2015, no license to the '397 patent had been
3 granted.

4 91. Plaintiff has been harmed by Defendant's infringing activities.

5 **COUNT II – INFRINGEMENT OF U.S. PATENT NO. 7,594,168**

6 92. The allegations set forth in the foregoing paragraphs 1 through 91 are incorporated
7 into this Second Claim for Relief.

8 93. On September 22, 2009, U.S. Patent No. 7,594,168 entitled *Browser Based Web Site*
9 *Generation Tool and Run Time Engine* was duly and legally issued by the United States Patent and
10 Trademark Office. A true and correct copy of the '168 patent is attached as Exhibit E.

11 94. The inventions of the '168 patent resolve technical problems related to website
12 creation and generation. For example, the inventions enable the creation of websites through
13 browser-based build tools and a user interface, which features are exclusively implemented utilizing
14 computer technology.

15 95. The claims of the '168 patent do not merely recite the performance of some business
16 practice known from the pre-Internet world along with the requirement to perform it on the Internet.
17 Instead, the claims of the '168 patent recite one or more inventive concepts that are rooted in
18 computerized website creation technology, and overcome problems specifically arising in the realm
19 of computerized website creation technologies.

20 96. The claims of the '168 patent recite an invention that is not merely the routine or
21 conventional use of website creation systems and methods. Instead, the invention describes a
22 browser-based website creation system including a server comprising a build engine configured to
23 create and apply styles to, for example, a website with web pages comprised of objects.

24 97. The technology claimed in the '168 patent does not preempt all ways of using website
25 or web page authoring tools nor preempt the use of all website or web page authoring tools, nor
26 preempt any other well-known or prior art technology.

27
28

1 98. Accordingly, each claim of the '168 patent recites a combination of elements
2 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an
3 ineligible concept.

4 99. As noted above and incorporated into this Second Claim for Relief, defendants in
5 other cases in which the '397 and '168 patents were asserted, asserted that the '397 and '168 patents
6 were invalid under 35 U.S.C. § 101. Those motions and related Orders are discussed above.

7 100. Plaintiff is the assignee and owner of the right, title and interest in and to the '168
8 patent, including the right to assert all causes of action arising under said patents and the right to any
9 remedies for infringement of them.

10 101. Upon information and belief, Defendant has and continues to directly infringe at least
11 claims 1-6 of the '168 patent by using a browser-based website and/or web page authoring tool in
12 which the user-selected settings representing website elements are stored in a database, and retrieval
13 of said information to generate said website (the "Accused Instrumentalities"). The Accused
14 Instrumentalities include but are not limited website building tools used and/or provided by
15 Defendant, such as, for example Drupal, Magento, Shopify and/or Wordpress. *See, e.g.*,
16 <https://www.rishabhsoft.com/web-application/drupal-development-services>;
17 <https://www.rishabhsoft.com/magento-development-services>; [https://www.rishabhsoft.com/web-](https://www.rishabhsoft.com/web-application/ecommerce-development)
18 [application/ecommerce-development](https://www.rishabhsoft.com/web-application/portals-collaboration); [https://www.rishabhsoft.com/web-application/portals-](https://www.rishabhsoft.com/web-application/portals-collaboration)
19 [collaboration](https://www.rishabhsoft.com/web-application/portals-collaboration).

20 102. In particular, claim 1 of the '168 patent generally recites a system for assembling a
21 website comprising a server with a build engine, the website comprising web pages with objects (one
22 button or one image object), the server accepting user input to associate a style with objects, wherein
23 a button or image object is associated with a style that includes values defining transformations and
24 time lines; wherein each web page is defined entirely by the objects and the style associated with the
25 object, produce a database with a multidimensional array comprising the objects that comprise the
26 website including data defining the object style, number, and an indication of the web page that each
27 object is part of, and provide the database to a server accessible to web browser; wherein the
28

1 database is produced such that a web browser with access to a runtime engine is configured to
2 generate the website from the objects and style data extracted from the provided database.

3 103. The Accused Instrumentalities infringe claim 1 of the '168 patent through a
4 combination of features which collectively practice each limitation of claim 1. (*See, e.g.*,
5 https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model,
6 http://www.w3schools.com/js/js_htmldom.asp.)

7 104. Further, by way of example, the JSON strings that are used by the Accused
8 Instrumentalities to generate, in part, element formatting originate from the database and therefore
9 reflect the database structure and contents showing, on information and belief, the implementation of
10 a multidimensional array structured database comprising the objects that comprise the web site. By
11 way of further evidence, the JSON strings show that there are dimensions for the pages, for arrays of
12 columns, for arrays of sections, and for arrays of modules generated using the Accused
13 Instrumentalities. *See, e.g.*,

14 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>

15 105. Further, the Accused Instrumentalities enable the storing in the database of data
16 defining each object such as object styles, an object number, and an indication of the which page
17 each object is a part of. For example, a user can select a theme style for a body title on a specific
18 page. The CSS database file is thereafter saved to the server, reflecting the selected font, size, and
19 the object and page to which it applies.

20 106. By way of example, for the completed web site, the Accused Instrumentalities include
21 runtime files, such as, for example HTML CSS files. *See, e.g.*, <https://www.drupal.org/home>;
22 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
23 <https://www.drupal.org/project/ckeditor>;
24 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
25 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
26 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
27 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
28 <https://www.drupal.org/docs/8/system-requirements/web-server>;

1 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
2 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
3 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
4 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
5 <https://www.drupal.org/docs/8/core/modules/media/overview>;
6 https://www.drupal.org/project/media_entity;
7 <https://www.drupal.org/docs/8/core/modules/image/working-with-images>;
8 <http://demos.dojotoolkit.org/demos/css3/demo.html>; <https://www.drupal.org/files/issues/Field.png>;
9 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>.;
10 https://www.drupal.org/project/save_draft.

11 107. Claim 2 of the '168 patent generally recites the system of claim 1, wherein one of
12 said plurality of objects is a child, and wherein the build engine is configured to accept user input to
13 associate a style with child button and child image objects.

14 108. The Accused Instrumentalities infringe claim 2 of the '168 patent through a
15 combination of features which collectively practice each limitation of claim 2.

16 109. By way of example from Express Mobile's investigatory use of the Accused
17 Instrumentalities, users of the Accused Instrumentalities are able to access child element images by
18 clicking on a parent element which unlock additional images related to a product being viewed.

19 110. Claim 3 of the '168 patent generally recites the system of claim 2, wherein at least
20 one of said styles includes values defining timelines for child button and child image objects.

21 111. The Accused Instrumentalities infringe claim 3 of the '168 patent through a
22 combination of features which collectively practice each limitation of claim 3.

23 112. By way of example, the Accused Instrumentalities incorporate various CSS libraries,
24 and CSS-animations and CSS-transitions are used extensively for adding transformations and
25 timelines to selected elements. On information and belief, this includes timelines for child buttons
26 and child image objects. *See* http://docs.magento.com/m1/ee/user_guide/cms/banner-rotator.html?Highlight=carousel.
27
28

1 113. Claim 4 of the '168 patent generally recites the system of claim 1, wherein at least
2 one of said styles includes settings for multiple object states.

3 114. The Accused Instrumentalities infringe claim 4 of the '168 patent through a
4 combination of features which collectively practice each limitation of claim 4.

5 115. By way of example, the Accused Instrumentalities enable the ability to define a hover
6 state, so that an element, including a button, has defined styles. *See, e.g.*,
7 <https://www.drupal.org/docs/8/core/themes/seven-theme>.

8 116. Claim 5 of the '168 patent generally recites the system of claim 1, further including
9 file size reduction means for reducing total size of files generated by said build engine to a size
10 between 12k and 50k.

11 117. On information and belief, the Accused Instrumentalities infringe claim 5 of the '168
12 patent through a combination of features which collectively practice each limitation of claim 5. *See*
13 http://docs.magento.com/m1/ee/user_guide/design/merge-css.html?Highlight=css.

14 118. Claim 6 of the '168 patent generally recites the system of claim 1, where said data is
15 stored as one or more of a Boolean an integer, a string, a floating point variables, or a URL.

16 119. The Accused Instrumentalities infringe claim 6 of the '168 patent through a
17 combination of features which collectively practice each limitation of claim 6. A review of the API
18 documentation behind websites created using the Accused Instrumentalities reveals data that is
19 stored as one or more of a Boolean, an integer, or a string. *See, e.g.*,
20 <https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions>.

21 120. Upon information and belief, these Accused Instrumentalities are used, marketed,
22 provided to, and or used by or for each of Defendant's partners, clients, customers, and/or end users
23 across the country and in this District.

24 121. In particular, Defendant's actions that aid and abet others such as its partners,
25 customers, clients, and/or end users to infringe include advertising and distributing the Accused
26 Instrumentalities and providing instruction materials, training, and services regarding the Accused
27 Instrumentalities. *See, e.g.*, <https://www.rishabhsoft.com/blog/wordpress-or-drupal>. On information
28 and belief, Defendant has engaged in such actions with specific intent to cause infringement or with

1 willful blindness to the resulting infringement because Defendant has had actual knowledge of the
2 '168 patent and knowledge that its acts were inducing infringement of the '397 patent since at least
3 the date Rishabh received notice that such activities infringed the '168 patent.

4 122. Upon information and belief, Defendant is liable as a contributory infringer of the
5 '168 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
6 website authoring tools to be especially made or adapted for use in an infringement of the '168
7 patent. The Accused Instrumentalities are a material component for use in practicing the '168 patent
8 and are specifically made and are not a staple article of commerce suitable for substantial non-
9 infringing use.

10 123. Defendant was made aware of the '168 patent and its infringement thereof at least as
11 early as the filing of this Complaint.

12 124. Since the date of the filing of this Complaint, Defendant's infringement of the '168
13 patent has been willful.

14 125. Plaintiff has not sold any product nor offered a service within the scope of any claim
15 of the '168 patent. In addition, prior to August 12, 2015, no license to the '168 patent had been
16 granted.

17 126. Plaintiff has been harmed by Defendant's infringing activities.

18 **COUNT III – INFRINGEMENT OF U.S. PATENT NO. 9,471,287**

19 127. The allegations set forth in the foregoing paragraphs 1 through 126 are incorporated
20 into this Third Claim for Relief.

21 128. On October 18, 2016, U.S. Patent No. 9,471,287 ("the '287 patent"), entitled
22 "*Systems and Methods for Integrating Widgets on Mobile Devices*," was duly and legally issued by
23 the United States Patent and Trademark Office. A true and correct copy of the '287 patent is
24 attached as Exhibit F.

25 129. The inventions of the '287 patent resolve technical problems related to generating
26 content on a display of a device, such as the display of a mobile device. For example, the inventions
27 of the '287 patent feature a registry and an authoring tool or Player configured to define a User
28 Interface ("UI") object for display on the device, where the UI object corresponds to a web

1 component. Each UI object is either: 1) selected by a user or 2) automatically selected by the system
2 as a preferred UI object corresponding to a symbolic name of the web component and used to
3 produce an Application, where the Application is a device-independent code; and a Player, where the
4 Player is a device-dependent code. The Application and Player enable 1) the device to provide one
5 or more input values and corresponding input symbolic name to the web service and 2) the web
6 service to utilize the input symbolic name and the user provided one or more input values to generate
7 one or more output values having an associated output symbolic name, while 3) the Player receives
8 the output symbolic name and corresponding one or more output values and provide instructions for
9 the display of the device to present an output value in the defined UI object. These features are
10 exclusively implemented utilizing computer technology.

11 130. The claims of the '287 patent do not merely recite the performance of some business
12 practice known from the pre-Internet world along with the requirement to perform it on the Internet.
13 Instead, the claims of the '287 patent recite one or more inventive concepts that are rooted in the
14 computerized generation of content on a display of a device, such as a mobile device, and overcome
15 problems specifically arising in the realm of computerized display content generation technologies.

16 131. The claims of the '287 patent recite inventions that are not merely the routine or
17 conventional use of systems and methods for the computerized generation of content on a display of
18 a device. Instead, the inventions feature systems for use with devices and methods of using the
19 systems with authoring tools or Players specific to each device and Applications that are device
20 independent.

21 132. The technology claimed in the '287 patent does not preempt all ways for the
22 computerized generation of content on a display of a device, such as a mobile device, nor preempt
23 the use of all authoring tools or Players for the computerized generation of content on a display of a
24 device, such as a mobile devices, nor preempt any other well-known or prior art technology.

25 133. Accordingly, each claim of the '287 patent recites a combination of elements
26 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an
27 ineligible concept.

28

1 134. Plaintiff is the assignee and owner of the right, title and interest in and to the '287
2 patent, including the right to assert all causes of action arising under the patents and the right to any
3 remedies for infringement of them.

4 135. Upon information and belief, Defendant has and continues to directly infringe at least
5 claims 1-5, 11, 12, 15-19, 25 and 26 of the '287 patent by a system and method which includes a
6 registry and an authoring tool or Player configured to define a User Interface ("UI") object for
7 display on the device, where the UI object corresponds to a web component. Each UI object is
8 either: 1) selected by a user or 2) automatically selected by the system as a preferred UI object
9 corresponding to a symbolic name of the web component and used to produce an Application, where
10 the Application is a device-independent code and a Player, where the Player is a device-dependent
11 code. The Application and Player enable 1) the device to provide one or more input values and
12 corresponding input symbolic name to the web service and 2) the web service to utilize the input
13 symbolic name and the user provided one or more input values to generate one or more output
14 values having an associated output symbolic name, while 3) the Player receives the output symbolic
15 name and corresponding one or more output values and provides instructions for the display of the
16 device to present an output value in the defined UI object (the "Accused Instrumentalities"). The
17 Accused Instrumentalities include platforms that enable the functionality described above and
18 include but are not limited to, for example, WordPress. *See, e.g.,* [https://www.rishabhsoft.com/web-](https://www.rishabhsoft.com/web-application/portals-collaboration)
19 [application/portals-collaboration](https://www.rishabhsoft.com/web-application/portals-collaboration).

20 136. In particular, claim 1 of the '287 patent recites 1 a system for generating code to
21 provide content on a display of a device, the system comprising: computer memory storing a registry
22 of: a) symbolic names required for evoking one or more web components each related to a set of
23 inputs and outputs of a web service obtainable over a network, where the symbolic names are
24 character strings that do not contain either a persistent address or pointer to an output value
25 accessible to the web service, where each symbolic name has an associated data format class type
26 corresponding to a subclass of User Interface (UI) objects that support the data format type of the
27 symbolic name, and has a preferred UI object, and b) an address of the web service; an authoring
28 tool configured to: define a (UI) object for presentation on the display, where the defined UI object

1 corresponds to a web component included in the registry selected from a group consisting of an input
2 of the web service and an output of the web service, where each defined UI object is either: 1)
3 selected by a user of the authoring tool; or 2) automatically selected by the system as the preferred
4 UI object corresponding to the symbolic name of the web component selected by the user of the
5 authoring tool, access the computer memory to select the symbolic name corresponding to the web
6 component of the defined UI object, associate the selected symbolic name with the defined UI
7 object, where the selected symbolic name is only available to UI objects that support the defined
8 data format associated with that symbolic name, and produce an Application including the selected
9 symbolic name of the defined UI object, where the Application is a device-independent code; and a
10 Player, where the Player is a device-dependent code, wherein, when the Application and Player are
11 provided to the device and executed on the device, and when the user of the device provides one or
12 more input values associated with an input symbolic name to an input of the defined UI object, 1)
13 the device provides the user provided one or more input values and corresponding input symbolic
14 name to the web service, 2) the web service utilizes the input symbolic name and the user provided
15 one or more input values for generating one or more output values having an associated output
16 symbolic name, 3) the Player receives the output symbolic name and corresponding one or more
17 output values and provides instructions for the display of the device to present an output value in the
18 defined UI object.

19 137. The Accused Instrumentalities infringe claim 1 of the '287 patent through a
20 combination of features The Accused Instrumentalities infringe claim 1 of the '287 patent through a
21 combination of features which collectively practice each limitation of claim 1. By way of example,
22 the Accused Instrumentalities feature a system for generating code to provide content on a display of
23 a device. The system includes a server hosting the WordPress platform, which provides
24 WordPress's WYSIWYG visual effects editor, and which is accessed through a WordPress-
25 compatible browser. WordPress's WYSIWYG visual effects editor generates code, such as
26 JavaScript or HTML code, for such options as defining title, text, images, videos and paragraph
27 styles, while the browser displays the resulting content as a WordPress webpage on a display of a
28 device.

138. For example, on information and belief, WordPress uses a variety of databases in its technology stack including MySQL. Data from the wp_options table for the website header and the data from the wp_posts table for the “WordPress Info” web page extracted directly from the Bitnami WordPress server-side database using MySQL Workbench. The stored data in the wp_options table includes the website’s url, the website’s title (blogname), the website’s tagline (blogdescription), and the active template (style sheet).

option_id	option_name	option_value
1	siteurl	http://localhost/wordpress
2	blogname	Express Mobile
3	blogdescription	SELECTED USERS OF WORDPRESS
44	template	the-fundamentals-of-graphic-design

The stored data in the wp_posts table for the “WordPress Info” web page includes information corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages 22%”. Other user selections shown in stored database data below include, for ex-ample, the image filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the image alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for “Heading 3”.

ID	post_content	post_title	post_type
23	<p>WordPress is used by over 14.7% of Alexa Internet's "top 1 million" websites and as of August, 2011, believe it or not, manages 22% of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&nbsp;</p> <p style="text-align: center;"></p></p> <p>&nbsp;</p> <p><h3>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</h3></p> <p>&nbsp;</p>	WordPress Info	page

1 WordPress's HTML, CSS, Java, and JSON coding capabilities further are shown, *e.g.*, by
2 http://codex.wordpress.org/Templates_, <https://codex.wordpress.org/CSS>,
3 <https://developer.wordpress.org/rest-api/>.

4 139. The Accused Instrumentalities feature a computer memory provided by WordPress
5 MySQL database functionality on the WordPress server. By way of example, WordPress utilizes
6 JSON strings extensively as part of its API, which necessarily require servers and databases. (*See*,
7 <https://developer.wordpress.org/rest-api/>.)

8 140. The computer memory stores a registry of a) symbolic names required for evoking
9 one or more web components each related to a set of inputs and outputs of a web service obtainable
10 over a network, where the symbolic names are character strings that do not contain either a persistent
11 address or pointer to an output value accessible to the web service. The WordPress MySQL
12 database contains symbolic names required for evoking one or more web components each related to
13 a set of inputs and outputs of a web service obtainable over a network by the formatting of the
14 symbolic names in conjunction with WordPress's WYSIWYG visual effects editor, widget, and
15 plugin authoring tools.

16 141. Furthermore, each symbolic name has an associated data format class type
17 corresponding to a subclass of defined UI objects *i.e.*, element/UI components, that supports the data
18 format type of the symbolic name, and has a preferred UI object as evidenced by the JSON
19 formatting of the name in conjunction with WordPress' WYSIWYG visual effects editor and widget
20 authoring tools. JSON names are strings that only represent the symbolic names that are bound both
21 to a web service input and/or output and to a UI object. All JSON names in the name/value pairs are
22 character strings. WordPress' WYSIWYG visual effects editor includes elements for defining the
23 layout for placement of the defined UI objects. Widgets, plug-ins and other elements correspond to
24 the defined UI objects and are the product of the JSON formatting. (*See*, for example,
25 https://codex.wordpress.org/WordPress_Lessons#Template_Files_,
26 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
27 and https://codex.wordpress.org/Plugin_API.)

28

1 142. The computer memory also stores b) an address of the web service. Because
2 WordPress contains web services, it contains the corresponding addresses for the web services.
3 (See, for example, <https://developer.wordpress.org/rest-api/>,
4 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
5 and https://codex.wordpress.org/Plugin_API.)

6 143. The Accused Instrumentalities feature an authoring tool in the form of WordPress's
7 WYSIWYG visual effects editor, widget, and plug-in authoring tools. (See, for example,
8 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
9 https://codex.wordpress.org/Plugin_API.)

10 144. The authoring tool is configured to define a UI object for presentation on the display,
11 where the defined UI object corresponds to a web component included in the registry selected from a
12 group consisting of an input of the web service and an output of the web service. WordPress's
13 WYSIWYG visual effects editor and widget authoring tools define the presence of a defined UI
14 object for presentation on a display and the defined UI object corresponds to a web component
15 included in the computer memory selected from a group consisting of an input of the web service
16 and an output of the web service.

17 145. Each defined UI object is either: 1) selected by a user of the authoring tool; or 2)
18 automatically selected by the system as the preferred UI object corresponding to the symbolic name
19 of the web component selected by the user of the authoring tool. WordPress's UI objects are
20 automatically selected by the system as the preferred UI object corresponding to the symbolic name
21 of the web component selected by the user of the authoring tool, i.e., a UI object selected by a user is
22 automatically selected. When a Widget is selected in the WordPress Widget selection list steps 1 to 5
23 under "Displaying Widgets" the widget UI will automatically displayed in the Web Page Sidebar.
24 (See, for example, https://codex.wordpress.org/WordPress_Widgets,
25 https://codex.wordpress.org/Plugin_Resources, https://codex.wordpress.org/Plugin_API.)

26 146. The authoring tool is configured to access the computer memory to select the
27 symbolic name corresponding to the web component of the defined UI object by a JSON formatted
28 element.

1 147. The authoring tool is also configured to associate the selected symbolic name with the
2 defined UI object, i.e., the JSON formatted element, where the selected symbolic name is only
3 available to UI objects that support the defined data format associated with the element associated
4 with that symbolic name, i.e., JSON string. JSON names are strings that only represent the symbolic
5 names that are bound both to a web service input and/or output and to a UI object. All JSON names
6 in the name/value pairs are character strings. When the WordPress Editor makes a UI element
7 request a JSON request is sent to the Server and a JSON data response is provided to the Interface.
8 (*See*, for example, <https://developer.wordpress.org/rest-api/>,
9 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
10 https://codex.wordpress.org/Plugin_API.)

11 148. The WordPress authoring tool is configured to produce an Application including the
12 selected symbolic name of the defined UI object, (*see*, for example,
13 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
14 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API), where
15 the Application is a device-independent with its API and “responsive” capabilities. (*See*, for
16 example, <https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/> and
17 <https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/>.)

18 149. The WordPress authoring tool is further configured to produce a Player (*see*, for
19 example, <https://codex.wordpress.org/CSS>, https://codex.wordpress.org/Using_Javascript,
20 <https://developer.wordpress.org/rest-api/>), where the Player is a device-dependent code. WordPress
21 contains a Player in the form of a runtime player. The Accused Instrumentality produces a device
22 dependent file, which is wrapped inside the runtime file. In order for a site to display on different
23 devices through a browser or through responsive capabilities, there is device dependent code (*see*,
24 for example, <https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/> and
25 <https://wordpress.org/themes/ultra/>.)

26 150. The Accused Instrumentalities feature a system where the Application and Player are
27 provided to the device and executed on the device and when the user of the device provides one or
28 more input values associated with an input symbolic name to an input of the defined UI object.

1 Because the Accused Instrumentalities incorporate a system that includes WordPress, when a user of
2 the device provides one or more input values associated with an input symbolic name, using JSON
3 formatting characteristics, to an input of the defined UI object, the device provides the user provided
4 one or more input values and corresponding input symbolic name, using JSON formatting
5 characteristics, to the web service. (See, for example, <https://developer.wordpress.org/rest-api/>,
6 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
7 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API.)

8 151. The Accused Instrumentalities feature a system where the device provides the user
9 provided one or more input values and corresponding input symbolic name to the web service.

10 Because the Accused Instrumentalities incorporate a system that includes WordPress, the web
11 service utilizes the input symbolic name and the user provided one or more input values for
12 generating one or more output values having an associated output symbolic name. The defined UI
13 object output value corresponds to the output symbolic name based on its JSON formatting
14 characteristics. (See, for example, <https://developer.wordpress.org/rest-api/>,
15 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
16 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API.)

17 152. The Accused Instrumentalities feature a system where the web service utilizes the
18 input symbolic name and the user provided one or more input values for generating one or more
19 output values having an associated output symbolic name. Because of the JSON formatting, the
20 output values having an associated output symbolic name. (See, for example,
21 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
22 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
23 https://codex.wordpress.org/Plugin_API.)

24 153. The Accused Instrumentalities feature a system where the Player receives the output
25 symbolic name and corresponding one or more output values and provides instructions for the
26 display of the device to present an output value in the defined UI object. The runtime player within
27 WordPress receives the output name, output value, and provides instructions for a display as shown
28 by the fact that the defined UI object are ultimately rendered. (See, for example,

1 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
2 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
3 https://codex.wordpress.org/Plugin_API.)

4 154. The presence of the above referenced features is demonstrated, by way of example,
5 by reference to publicly available information. Regarding WordPress, *see*, e.g.,
6 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
7 http://codex.wordpress.org/Template_Hierarchy;
8 http://codex.wordpress.org/Function_Reference/the_title;
9 http://codex.wordpress.org/Function_Reference/the_content;
10 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
11 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
12 http://codex.wordpress.org/Query_Overview.

13 155. Claim 2 of the '287 patent recites a system for generating code to provide content on
14 a display of a device that includes all the elements of claim 1, additionally where the registry
15 includes definitions of input and output related to the web service.

16 156. The Accused Instrumentalities infringe claim 2 of the '287 patent through a
17 combination of features which collectively practice each limitation of claim 1. By way of example,
18 the registry includes definitions of input and output related to a web service as evidenced by
19 WordPress's JSON formatting characteristics of the defined UI objects. (*See*, for example,
20 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
21 <https://developer.wordpress.org/rest-api/reference/posts/> <https://developer.wordpress.org/rest-api/reference/posts/#schema-title>, https://codex.wordpress.org/Widgets_API;
22 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
23 https://codex.wordpress.org/WordPress_Widgets.)
24

25 157. Claim 3 of the '287 patent recites a system for generating code to provide content on
26 a display of a device that includes all the elements of claim 1, additionally where the web component
27 is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.
28

1 158. The Accused Instrumentalities infringe claim 3 of the '287 patent through a
2 combination of features which collectively practice each limitation of claim 3. By way of example,
3 the Accused Instrumentalities feature web components additionally including web chat, Reuters RSS
4 feed, Calendar image, and map image widgets.

5 159. Claim 4 of the '287 patent recites a system for generating code to provide content on
6 a display of a device that includes all the elements of claim 1, additionally where the defined UI
7 object is an input field for a chat.

8 160. The Accused Instrumentalities infringe claim 4 of the '287 patent through a
9 combination of features which collectively practice each limitation of claim 4. By way of example,
10 the Accused Instrumentalities additionally feature a defined UI object that is an input field for a chat.

11 161. Claim 5 of the '287 patent recites a system for generating code to provide content on
12 a display of a device that includes all the elements of claim 1, additionally the defined UI object is an
13 input field for a web service.

14 162. The Accused Instrumentalities infringe claim 5 of the '287 patent through a
15 combination of features which collectively practice each limitation of claim 1. By way of example,
16 the Accused Instrumentalities additionally feature a defined UI object that is an input field for a web
17 service.

18 163. Claim 11 of the '287 patent recites a system for generating code to provide content on
19 a display of a device that includes all the elements of claim 1, additionally where the code is
20 provided over the network.

21 164. The Accused Instrumentalities infringe claim 11 of the '287 patent through a
22 combination of features which collectively practice each limitation of claim 11. WordPress sends all
23 files over a network using a variety of databases in its technology stack including MySQL. These
24 backend capabilities provided the code over a network. By way of example, data from the
25 wp_options table for the website header and from the wp_posts table for the "WordPress Info" web
26 page are extracted directly from the Bitnami WordPress server-side database using MySQL
27 Workbench. The stored data in the wp_options table includes the website's url, the website's title
28 (blogname), the website's tagline (blogdescription), and the active template (style sheet).

option_id	option_name	option_value
1	siteurl	http://localhost/wordpress
2	blogname	Express Mobile
3	blogdescription	SELECTED USERS OF WORDPRESS
44	template	the-fundamentals-of-graphic-design

The stored data in the wp_posts table for the “WordPress Info” web page includes information corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages 22%”. Other user selections shown in stored database data below include, for ex-ample, the image filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the image alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for “Heading 3”.

ID	post_content	post_title	post_type
23	<p>WordPress is used by over 14.7% of Alexa Internet's "top 1 million" websites and as of August, 2011, believe it or not, manages 22% of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&nbsp;</p> <p style="text-align: center;"></p></p> <p>&nbsp;</p> <p><h3>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</h3></p> <p>&nbsp;</p>	WordPress Info	page

WordPress’s HTML, CSS, Java, and JSON coding capabilities further are shown, for example. by http://codex.wordpress.org/Templates_, <https://codex.wordpress.org/CSS>, <https://developer.wordpress.org/rest-api/> and https://codex.wordpress.org/Widgets_API).

1 165. Claim 12 of the '287 patent recites a system for generating code to provide content
2 on a display of a device that includes all the elements of claim 1, additionally where the defined UI
3 object corresponds to a widget.

4 166. The Accused Instrumentalities infringe claim 12 of the '287 patent through a
5 combination of features which collectively practice each limitation of claim 1. By way of example,
6 the Accused Instrumentalities' incorporation of WordPress includes widgets. (See, for example.,
7 https://codex.wordpress.org/Widgets_API.)

8 167. Claim 15 of the '287 patent recites a method of displaying content on a display of a
9 device having a Player, where the Player is a device-dependent code, the method comprising:
10 defining a user interface (UI) object for presentation on the display, where the UI object corresponds
11 to a web component included in a registry of one or more web components selected from a group
12 consisting of an input of a web service and an output of the web service, where each web component
13 includes a plurality of symbolic names of inputs and outputs associated with each web service, and
14 where the registry includes: a) symbolic names required for evoking one or more web components
15 each related to a set of inputs and outputs of the web service obtainable over a network, where the
16 symbolic names are character strings that do not contain either a persistent address or pointer to an
17 output value accessible to the web service, and b) an address of the web service, and where each
18 defined UI object is either: 1) selected by a user of an authoring tool; 2) automatically selected by a
19 system as a preferred UI object corresponding to a symbolic name of the web component selected by
20 the user of the authoring tool.

21 168. The Accused Instrumentalities infringe claim 15 of the '287 patent through a
22 combination of features which collectively practice each limitation of claim 15. By way of example,
23 the method is for displaying content on a display of a device and the Accused Instrumentalities
24 include a WordPress-compatible browser which content as a WordPress webpage on a display of a
25 device.

26 169. The Accused Instrumentalities feature a Player, where the Player is a device-
27 dependent code. The device has a Player (*see*, for example, https://codex.wordpress.org/CSS_
28 https://codex.wordpress.org/Using_Javascript, <https://developer.wordpress.org/rest-api/>) in the form

1 of a runtime player that is a device dependent code. WordPress produces a device dependent file,
2 which is wrapped inside a runtime file. In order for a site to display on different devices through a
3 browser or through responsive capabilities, there is device dependent code. (*See*, for example,
4 <https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/>;
5 <https://wordpress.org/themes/ultra/>.)

6 170. The Accused Instrumentalities include defining a user interface (UI) object for
7 presentation on the display, where the defined UI object corresponds to a web component included
8 in the registry of one or more web components, where the web component is selected from a group
9 consisting of an input of a web service and an output of the web service. WordPress defines a UI
10 object for presentation on display, where the UI object corresponds to a web component included in
11 the non-volatile computer memory selected from a group consisting of an input of a web service and
12 an output of the web service by JSON data formatting. (*See*, for example,
13 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
14 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_API, and
15 https://codex.wordpress.org/Plugin_Resources.) JSON names are strings that only represent the
16 symbolic names that are bound both to a web service input and/or output and to a UI object. All
17 JSON names in the name/value pairs are character strings. When the WordPress interface makes a
18 UI element request a JSON request is sent to the Server and a JSON data response is provided to the
19 Interface. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
20 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
21 https://codex.wordpress.org/Plugin_API.)

22 171. Each web component includes a plurality of symbolic names of inputs and outputs
23 associated with each web service. The plurality of symbolic names of inputs and outputs associated
24 with each web service is a feature of their JSON formatting characteristics. Each symbolic name has
25 an associated data format class type corresponding to a subclass of UI objects that supports the data
26 format type of the symbolic name, and has a preferred UI object as demonstrated by the presence of
27 JSON formatting in conjunction with WordPress' WYSIWYG and widget capabilities. (*See*, for
28 example, <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/>,

1 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_API, and
2 https://codex.wordpress.org/Plugin_Resources.)

3 172. The registry includes: a) symbolic names required for evoking one or more web
4 components each related to a set of inputs and outputs of a web service obtainable over a network,
5 where the symbolic names are character strings that do not contain either a persistent address or
6 pointer to an output value accessible to the web service. The registry and WordPress MySQL
7 database contain symbolic names required for evoking one or more web components each related to
8 a set of inputs and outputs of a web service obtainable over a network as demonstrated by the
9 formatting in conjunction with WordPress's WYSIWYG visual effects editor widget, and plugin
10 authoring tools. JSON names are strings that only represent the symbolic names that are bound both
11 to a web service input and/or output and to a UI object. All JSON names in the name/value pairs are
12 character strings. When the WordPress interface makes a UI element request a JSON request is sent
13 to the Server and a JSON data response is provided to the Interface. (*See*, for example,
14 <https://developer.wordpress.org/rest-api/>, https://codex.wordpress.org/WordPress_Widgets,
15 https://codex.wordpress.org/Plugin_Resources, https://codex.wordpress.org/Plugin_API.)

16 173. The registry also includes b) an address of the web service. Because WordPress
17 contains web services, it contains the corresponding web addresses. Because WordPress contains
18 web services, both as a library of Widgets (*see* https://codex.wordpress.org/Widgets_API) and Plug-
19 ins (*see* https://codex.wordpress.org/Plugin_API), it contains the corresponding addresses of the web
20 services. All web services are represented as a wsdl (Web Service Description Language) and wsdls
21 have URLs to point to the internet location that receives the web service's inputs and returns the web
22 service's outputs (*see* <https://www.soapui.org/soap-and-wsdl/working-with-wsdls.html>).

23 174. Each defined UI object is either: 1) selected by a user of an authoring tool; or 2)
24 automatically selected by the system as the preferred UI object corresponding to the symbolic name
25 of the web component selected by the user of the authoring tool. WordPress's UI objects are
26 automatically selected by the system as the preferred UI object corresponding to the symbolic name
27 of the web component selected by the user of the authoring tool, i.e., a UI object selected by a user is
28 automatically selected. When a Widget is selected in the WordPress Widget selection list steps 1 to 5

1 under “Displaying Widgets“, the widget UI will automatically displayed in the Web Page Sidebar.
2 (See, for example, https://codex.wordpress.org/WordPress_Widgets,
3 https://codex.wordpress.org/Plugin_Resources, https://codex.wordpress.org/Plugin_API.)

4 175. The Accused Instrumentalities include selecting the symbolic name from the web
5 component (i.e. WordPress Widget or Plug-in) corresponding to the defined UI object, where the
6 selected symbolic name has an associated data format class type corresponding to a subclass of UI
7 objects that support the data format type of the symbolic name and has the preferred UI object.
8 WordPress accesses its memory to select the symbolic name corresponding to the web component of
9 the defined UI object (as evidenced by JSON data formatting), associate the selected symbolic name
10 with the defined UI object (the JSON element corresponding to an element), where the selected
11 symbolic name is only available to UI objects that support the defined data format associated with
12 that symbolic name (the element associated with at JSON string). (See, for example,
13 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
14 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_API, and
15 https://codex.wordpress.org/Plugin_Resources.) Additionally, the preferred UI object is the selected
16 UI object. JSON names are strings that only represent the symbolic names that are bound both to a
17 web service input and/or output and to a UI object. All JSON names in the name/value pairs are
18 character strings. When the WordPress interface makes a UI element request a JSON request is sent
19 to the Server and a JSON data response is provided to the Interface. (See, for example,
20 <https://developer.wordpress.org/rest-api/>, https://codex.wordpress.org/WordPress_Widgets,
21 https://codex.wordpress.org/Plugin_Resources, https://codex.wordpress.org/Plugin_API.)

22 176. The Accused Instrumentalities include associating the selected symbolic name with
23 the defined UI object.

24 177. The Accused Instrumentalities include producing an Application including the
25 selected symbolic name of the defined UI object, where the Application is a device-independent
26 code. WordPress produces an Application included in the symbolic name of the defined UI object.
27 (See, for example, <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest->
28

1 api/reference/, https://codex.wordpress.org/WordPress_Widgets,
2 https://codex.wordpress.org/Plugin_API, and https://codex.wordpress.org/Plugin_Resources.)

3 178. The Accused Instrumentalities feature a system where the Application and Player are
4 provided to the device and executed on the device and when the user of the device provides one or
5 more input values associated with an input symbolic name to an input of the defined UI object.
6 Because the Accused Instrumentalities incorporate a system that includes WordPress, when a user of
7 the device provides one or more input values associated with an input symbolic name, using JSON
8 formatting characteristics, to an input of the defined UI object, the device provides the user provided
9 one or more input values and corresponding input symbolic name, using JSON formatting
10 characteristics, to the web service. (See, for example, <https://developer.wordpress.org/rest-api/>,
11 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
12 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API.)

13 179. The Accused Instrumentalities feature a system where the device provides the user
14 provided one or more input values and corresponding input symbolic name to the web service.
15 Because the Accused Instrumentalities incorporate a system that includes WordPress, the web
16 service utilizes the input symbolic name and the user provided one or more input values for
17 generating one or more output values having an associated output symbolic name. The defined UI
18 object output value corresponds to the output symbolic name based on its JSON formatting
19 characteristics. (See, for example, <https://developer.wordpress.org/rest-api/>,
20 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
21 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API.)

22 180. The Accused Instrumentalities feature a system where the web service utilizes the
23 input symbolic name and the user provided one or more input values for generating one or more
24 output values having an associated output symbolic name. Because of the JSON formatting, the
25 output values having an associated output symbolic name. (See, for example,
26 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
27 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
28 https://codex.wordpress.org/Plugin_API.)

1 181. The Accused Instrumentalities feature a system where the Player receives the output
2 symbolic name and corresponding one or more output values and provides instructions for the
3 display of the device to present an output value in the defined UI object. The runtime player within
4 WordPress receives the output name, output value, and provides instructions for a display as shown
5 by the fact that the defined UI object are ultimately rendered. (*See*, for example,
6 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
7 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
8 https://codex.wordpress.org/Plugin_API.)

9 182. The presence of the above referenced features is demonstrated, by way of example,
10 by reference to publicly available information. Regarding WordPress, *see, e.g.*,
11 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
12 http://codex.wordpress.org/Template_Hierarchy;
13 http://codex.wordpress.org/Function_Reference/the_title;
14 http://codex.wordpress.org/Function_Reference/the_content;
15 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
16 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
17 http://codex.wordpress.org/Query_Overview.

18 183. Claim 16 of the '287 patent recites a method of displaying content on a display that
19 includes all the elements of claim 15, additionally where the registry includes definitions of input
20 and output related to the web service.

21 184. The Accused Instrumentalities infringe claim 16 of the '287 patent through a
22 combination of features which collectively practice each limitation of claim 16. By way of example,
23 the Accused Instrumentalities include definitions of input and output related to a web service as
24 based on their JSON formatting characteristics and the defined UI object functionality. *See, e.g.*,
25 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
26 <https://developer.wordpress.org/rest-api/reference/posts/> [https://developer.wordpress.org/rest-](https://developer.wordpress.org/rest-api/reference/posts/#schema-title)
27 [api/reference/posts/#schema-title](https://developer.wordpress.org/rest-api/reference/posts/#schema-title) https://codex.wordpress.org/Widgets_API;
28

1 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
2 https://codex.wordpress.org/WordPress_Widgets.

3 185. Claim 17 of the '287 patent recites a method of displaying content on a display that
4 includes all the elements of claim 15, additionally where the web component is a text chat, a video
5 chat, an image, a slideshow, a video, or an RSS feed.

6 186. The Accused Instrumentalities infringe claim 17 of the '287 patent through a
7 combination of features which collectively practice each limitation of claim 17. By way of example,
8 the Accused Instrumentalities feature web components including web chat, Reuters RSS feed,
9 Calendar image, and map image widgets.

10 187. Claim 18 of the '287 patent recites a method of displaying content on a display that
11 includes all the elements of claim 15, additionally where the defined UI object is an input field for a
12 chat.

13 188. The Accused Instrumentalities infringe claim 18 of the '287 patent through a
14 combination of features which collectively practice each limitation of claim 18. By way of example,
15 the Accused Instrumentalities additionally feature an UI object that is an input field for a chat.

16 189. Claim 19 of the '287 patent recites a method of displaying content on a display that
17 includes all the elements of claim 15, additionally where the UI object is an input field for a web
18 service.

19 190. The Accused Instrumentalities infringe claim 19 of the '287 patent through a
20 combination of features which collectively practice each limitation of claim 19. By way of example,
21 the Accused Instrumentalities additionally feature a UI object that is an input field for a web service.

22 191. Claim 25 of the '287 patent recites a method of displaying content on a display that
23 includes all the elements of claim 15, additionally where the method includes providing the
24 Application and Player over the network.

25 192. The Accused Instrumentalities infringe claim 25 of the '287 patent through a
26 combination of features which collectively practice each limitation of claim 25. WordPress sends all
27 files over a network using a variety of databases in its technology stack including MySQL. These
28 backend capabilities provided the code over a network. By way of example, data from the

1 wp_options table for the website header and from the wp_posts table for the “WordPress Info” web
 2 page are extracted directly from the Bitnami WordPress server-side database using MySQL
 3 Workbench. The stored data in the wp_options table includes the website’s url, the website’s title
 4 (blogname), the website’s tagline (blogdescription), and the active template (style sheet).

option_id	option_name	option_value
1	siteurl	http://localhost/wordpress
2	blogname	Express Mobile
3	blogdescription	SELECTED USERS OF WORDPRESS
44	template	the-fundamentals-of-graphic-design

5
 6
 7
 8
 9
 10 The stored data in the wp_posts table for the “WordPress Info” web page includes information
 11 corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages
 12 22%”. Other user selections shown in stored database data below include, for ex-ample, the image
 13 filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the
 14 image alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for
 15 “Heading 3”.

ID	post_content	post_title	post_type
23	<p>WordPress is used by over 14.7% of Alexa Internet's "top 1 million" websites and as of August, 2011, believe it or not, manages 22% of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&nbsp;</p> <p></p></p> <p>&nbsp;</p> <p><h3>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</h3></p> <p>&nbsp;</p>	WordPress Info	page

1 WordPress's HTML, CSS, Java, and JSON coding capabilities further are shown, for example by
2 <http://codex.wordpress.org/Templates>, <https://codex.wordpress.org/CSS>,
3 <https://developer.wordpress.org/rest-api/> and https://codex.wordpress.org/Widgets_API.

4 193. Claim 26 of the '287 patent recites a method of displaying content on a display that
5 includes all the elements of claim 15, additionally where the UI object corresponds to a widget.

6 194. The Accused Instrumentalities infringe claim 26 of the '287 patent through a
7 combination of features which collectively practice each limitation of claim 26. By way of example,
8 the Accused Instrumentalities employ WordPress which includes widgets. (*See*, for example.,
9 https://codex.wordpress.org/Widgets_API.)

10 195. Upon information and belief, these Accused Instrumentalities are used, marketed,
11 provided to, and or used by or for each of Defendant's partners, clients, customers, and/or end users
12 across the country and in this District.

13 196. In particular, Defendant's actions that aid and abet others such as its partners,
14 customers, clients, and/or end users to infringe include advertising and distributing the Accused
15 Instrumentalities and providing instruction materials, training, and services regarding the Accused
16 Instrumentalities. *See, e.g.*, <https://www.rishabhsoft.com/blog/wordpress-or-drupal>. On information
17 and belief, Defendant has engaged in such actions with specific intent to cause infringement or with
18 willful blindness to the resulting infringement because Defendant has had actual knowledge of the
19 '287 patent and knowledge that its acts were inducing infringement of the '287 patent since at least
20 the date Rishabh received notice that such activities infringed the '287 patent.

21 197. Upon information and belief, Defendant is liable as a contributory infringer of the
22 '287 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
23 website authoring tools to be especially made or adapted for use in an infringement of the '287
24 patent. The Accused Instrumentalities are a material component for use in practicing the '287 patent
25 and are specifically made and are not a staple article of commerce suitable for substantial non-
26 infringing use.

27 198. Defendant was made aware of the '287 patent and its infringement thereof at least as
28 early as the filing of this Complaint.

1 199. Since the date of the filing of this Complaint, Defendant’s infringement of the ’287
2 patent has been willful.

3 200. Within the past six years, Plaintiff has not sold any product nor offered a service
4 within the scope of any claim of the ’287 patent. In addition, prior to August 12, 2015, no license to
5 the ’287 patent had been granted.

6 201. Plaintiff has been harmed by Defendant’s infringing activities.

7 **COUNT IV – INFRINGEMENT OF U.S. PATENT NO. 9,928,044**

8 202. The allegations set forth in the foregoing paragraphs 1 through 201 are incorporated
9 into this Fourth Claim for Relief.

10 203. The allegations set forth in the foregoing paragraphs 1 through 138 are incorporated
11 into this Fourth Claim for Relief.

12 204. On March 27, 2018, U.S. Patent No. 9,928,044 (“the ’044 patent”), entitled “*Systems*
13 *and Methods for Programming Mobile Devices*,” was duly and legally issued by the United States
14 Patent and Trademark Office. A true and correct copy of the ’044 patent is attached as Exhibit G.

15 205. The inventions of the ’044 patent resolve technical problems related to generating
16 content on a display of a device, such as the display of a mobile device. For example, the inventions
17 feature a computer memory and an authoring tool or Player configured to define a User Interface
18 (“UI”) object for display on the device, where the defined UI object corresponds to a web
19 component and where each UI object is either: 1) selected by a user or 2) automatically selected by
20 the system as a preferred UI object corresponding to a symbolic name of the web component.
21 Additionally, the computer memory and the authoring tool or Player are configured to build an
22 Application consisting of one or more web page views to provide for the display of at least a portion
23 of one or more of the web pages. These features are exclusively implemented utilizing computer
24 technology.

25 206. The claims of the ’044 patent do not merely recite the performance of some business
26 practice known from the pre-Internet world along with the requirement to perform it on the Internet.
27 Instead, the claims of the ’044 patent recite one or more inventive concepts that are rooted in the
28

1 computerized generation of content on a display of a device, such as a mobile device, and overcome
2 problems specifically arising in the realm of computerized display content generation technologies.

3 207. The claims of the '044 patent recite an invention that is not merely the routine or
4 conventional use of systems and methods for the computerized generation of content on a display of
5 a device. Instead, the invention describes systems for use with devices with authoring tools or
6 Players specific to each device and Applications that are device independent.

7 208. The technology claimed in the '044 patent does not preempt all ways for the
8 computerized generation of content on a display of a device, such as a mobile device, nor preempt
9 the use of all authoring tools or Players for the computerized generation of content on a display of a
10 device, such as a mobile devices, nor preempt any other well-known or prior art technology.

11 209. Accordingly, each claim of the '044 patent recites a combination of elements
12 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an
13 ineligible concept.

14 210. Plaintiff is the assignee and owner of the right, title and interest in and to the '044
15 patent, including the right to assert all causes of action arising under the patents and the right to any
16 remedies for infringement of them.

17 211. Upon information and belief, Defendant has and continues to directly infringe at least
18 claims 1-5, 11, 12, 15-19, 25 and 26 of the '044 patent by a system which includes a computer
19 memory and an authoring tool or Player configured to define a User Interface (“UI”) object for
20 display on the device, where the UI object corresponds to a web component and where each UI
21 object is either: 1) selected by a user or 2) automatically selected by the system as a preferred UI
22 object corresponding to a symbolic name of the web component. Additionally, the computer
23 memory and the authoring tool or Player are configured to build an Application consisting of one or
24 more web page views to provide for the display of at least a portion of one or more of the web pages
25 (the “Accused Instrumentalities”). The Accused Instrumentalities include platforms that enable the
26 functionality described above and include but are not limited to, for example, WordPress. *See, e.g.,*
27 <https://www.rishabhsoft.com/web-application/portals-collaboration..>
28

1 212. In particular, claim 1 of the '044 patent recites a system for generating code to
2 provide content on a display of a device, the system comprising: computer memory storing: a)
3 symbolic names required for evoking one or more web components each related to a set of inputs
4 and outputs of a web service obtainable over a network, where the symbolic names are character
5 strings that do not contain either a persistent address or pointer to an output value accessible to the
6 web service, where each symbolic name has an associated data format class type corresponding to a
7 subclass of User Interface (UI) objects that support the data format type of the symbolic name, and
8 where each symbolic name has a preferred UI object, and b) an address of the web service; an
9 authoring tool configured to: define a UI object for presentation on the display, where the defined UI
10 object corresponds to a web component included in the computer memory selected from a group
11 consisting of an input of the web service and an output of the web service, where each defined UI
12 object is either: 1) selected by a user of the authoring tool; or 2) automatically selected by the system
13 as the preferred UI object corresponding to the symbolic name of the web component selected by the
14 user of the authoring tool, access the computer memory to select the symbolic name corresponding
15 to the web component of the defined UI object, associate the selected symbolic name with the
16 defined UI object, where the selected symbolic name is only available to UI objects that support the
17 defined data format associated with that symbolic name, store information representative of the
18 defined UI object and related settings in a database; retrieve the information representative of the
19 one or more the UI object settings stored in the database; and build an Application consisting of one
20 or more web page views from at least a portion of the database utilizing at least one Player, where
21 the Player utilizes information stored in the database to generate for the display of at least a portion
22 of the one or more web pages, wherein when the Application and Player are provided to the device
23 and executed on the device, and when the user of the device provides one or more input values
24 associated with an input symbolic name to an input of the defined UI object, the device provides the
25 user provided one or more input values and corresponding input symbolic name to the web service,
26 the web service utilizes the input symbolic name and the user provided one or more input values for
27 generating one or more output values having an associated output symbolic name, and the Player
28

1 receives the output symbolic name and corresponding one or more output values and provides
2 instructions for the display of the device to present an output value in the defined UI object.

3 213. The Accused Instrumentalities infringe claim 1 of the '044 patent through a
4 combination of features which collectively practice each limitation of claim 1. By way of example,
5 the Accused Instrumentalities feature a system for generating code to provide content on a display of
6 a device. The system includes a WordPress server, which provides WordPress's WYSIWYG visual
7 effects editor and a WordPress-compatible browser. WordPress's WYSIWYG visual effects editor
8 generates code, such as JavaScript or HTML code for such as options for defining title, text, images,
9 videos and paragraph styles, while the browser displays the resulting content as a WordPress
10 webpage on a display of a device, such as a computer display.

11 214. For example, on information and belief, WordPress uses a variety of databases in its
12 technology stack including MySQL. Data from the wp_options table for the website header and the
13 data from the wp_posts table for the "WordPress Info" web page extracted directly from the Bitnami
14 WordPress server-side database using MySQL Workbench. The stored data in the wp_options table
15 includes the website's url, the website's title (blogname), the website's tagline (blogdescription), and
16 the active template (style sheet).

option_id	option_name	option_value
1	siteurl	http://localhost/wordpress
2	blogname	Express Mobile
3	blogdescription	SELECTED USERS OF WORDPRESS
44	template	the-fundamentals-of-graphic-design

1 The stored data in the wp_posts table for the “WordPress Info” web page includes information
 2 corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages
 3 22%”. Other user selections shown in stored database data below include, for ex-ample, the image
 4 filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the
 5 image alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for
 6 “Heading 3”.

ID	post_content	post_title	post_type
23	<p>WordPress is used by over 14.7% of Alexa Internet's "top 1 million" websites and as of August, 2011, believe it or not, manages 22% of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&nbsp;</p> <p style="text-align: center;"></p></p> <p>&nbsp;</p> <p style="text-align: center;"><h3>as 2011,="" 3.0="" 65="" been="" december="" downloaded="" h3><="" had="" million="" of="" over="" p="" times.<="" version="" wordpress=""> <p>&nbsp;</p> </h3>as></p>	WordPress Info	page

18 WordPress’s HTML, CSS, Java, and JSON coding capabilities further are shown, *e.g.*, by
 19 http://codex.wordpress.org/Templates_, <https://codex.wordpress.org/CSS>,
 20 <https://developer.wordpress.org/rest-api/>.

21 215. The Accused Instrumentalities feature a computer memory provided by WordPress
 22 MySQL database functionality on the WordPress server. By way of example, WordPress utilizes
 23 JSON strings extensively as part of its API, which necessarily require servers and databases. (*See*,
 24 <https://developer.wordpress.org/rest-api/>.)

25 216. The computer memory stores a) symbolic names required for evoking one or more
 26 web components each related to a set of inputs and outputs of a web service obtainable over a
 27 network, where the symbolic names are character strings that do not contain either a persistent
 28

1 address or pointer to an output value accessible to the web service. The WordPress MySQL
2 database contains symbolic names required for evoking one or more web components each related to
3 a set of inputs and outputs of a web service obtainable over a network as demonstrated by the
4 formatting in conjunction with WordPress's WYSIWYG visual effects editor, widget, and plugin
5 authoring tools. JSON names are strings that only represent the symbolic names that are bound both
6 to a web service input and/or output and to a UI object. All JSON names in the name/value pairs are
7 character strings. WordPress' WYSIWYG visual effects editor includes elements for defining the
8 layout for placement of the defined UI objects. Widgets, plug-ins and other elements correspond to
9 the defined UI objects and are the product of the JSON formatting. (*See*, for example,
10 https://codex.wordpress.org/WordPress_Lessons#Template_Files,
11 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
12 and https://codex.wordpress.org/Plugin_API.)

13 217. Furthermore, each symbolic name has an associated data format class type
14 corresponding to a subclass of UI objects that supports the data format type of the symbolic name,
15 and has a preferred UI object as demonstrated by the presence of JSON formatting characteristics in
16 conjunction with WordPress' WYSIWYG visual effects editor and widget authoring tools. JSON
17 names are strings that only represent the symbolic names that are bound both to a web service input
18 and/or output and to a UI object. All JSON names in the name/value pairs are character strings.
19 WordPress' WYSIWYG visual effects editor includes elements for defining the layout for placement
20 of the defined UI objects. Widgets, plug-ins and other elements correspond to the defined UI objects
21 and are the product of the JSON formatting. (*See*, for example,
22 https://codex.wordpress.org/WordPress_Lessons#Template_Files,
23 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
24 and https://codex.wordpress.org/Plugin_API.)

25 218. The computer memory also stores b) an address of the web service. Because
26 WordPress contains web services, it contains the corresponding addresses for the web services. (*See*,
27 for example, <https://developer.wordpress.org/rest-api/>,
28

1 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
2 and https://codex.wordpress.org/Plugin_API.)

3 219. The Accused Instrumentalities feature an authoring tool in the form of WordPress's
4 WYSIWYG visual effects editor, widget, and plug-in authoring tools. (*See*, for example,
5 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
6 https://codex.wordpress.org/Plugin_API.)

7 220. The authoring tool is configured to define a UI object for presentation on the display,
8 where the defined UI object corresponds to a web component included in the computer memory
9 selected from a group consisting of an input of the web service and an output of the web service.
10 WordPress's WYSIWYG visual effects editor and widget authoring tools define the presence of a UI
11 object for presentation on a display and the defined UI object corresponds to a web component
12 included in the computer memory selected from a group consisting of an input of the web service
13 and an output of the web service.

14 221. Each defined UI object is either: 1) selected by a user of the authoring tool; or 2)
15 automatically selected by the system as the preferred UI object corresponding to the symbolic name
16 of the web component selected by the user of the authoring tool. WordPress's UI objects are
17 automatically selected by the system as the preferred UI object corresponding to the symbolic name
18 of the web component selected by the user of the authoring tool, *i.e.*, a UI object selected by a user is
19 automatically selected. When a Widget is selected in the WordPress Widget selection list (*See*
20 https://codex.wordpress.org/WordPress_Widgets) steps 1 to 5 under "Displaying Widgets", the
21 widget UI will automatically displayed in the Web Page Sidebar. (*See*, for example,
22 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
23 https://codex.wordpress.org/Plugin_API.)

24 222. The authoring tool is configured to access the computer memory to select the
25 symbolic name corresponding to the web component of the defined UI object based on its JSON
26 formatting characteristics.

27 223. The authoring tool is also configured to associate the selected symbolic name with the
28 defined UI object, *i.e.*, the JSON formatted element, where the selected symbolic name is only

1 available to UI objects that support the defined data format associated with the element associated
2 with that symbolic name, i.e., JSON string. JSON names are strings that only represent the symbolic
3 names that are bound both to a web service input and/or output and to a UI object. All JSON names
4 in the name/value pairs are character strings. When the WordPress Editor makes a UI element
5 request a JSON request is sent to the Server and a JSON data response is provided to the Interface.
6 (*See*, for example, <https://developer.wordpress.org/rest-api/>,
7 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
8 https://codex.wordpress.org/Plugin_API.)

9 224. The authoring tool is configured to store information representative of the defined UI
10 object and related settings in a database. For example, WordPress's computer memory is configured
11 to store information representative of defined UI objects. (*See*, for example,
12 <https://developer.wordpress.org/rest-api/> and https://codex.wordpress.org/Widgets_API.)

13 225. The authoring tool is also configured to retrieve the information representative of the
14 one or more the UI object settings stored in the database based on the JSON strings. (*See*, for
15 example, <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
16 https://codex.wordpress.org/Plugin_Resources,
17 https://codex.wordpress.org/Plugin_API, and https://codex.wordpress.org/Widgets_API.)

18 226. The WordPress authoring tool is configured to build an Application consisting of one
19 or more web page views from at least a portion of the database utilizing at least one Player, where
20 the Player utilizes information stored in the database to generate for the display of at least a portion
21 of the one or more web pages. WordPress builds an Application including the symbolic name of the
22 defined UI object. (*See*, for example, https://codex.wordpress.org/Widgets_API,
23 https://codex.wordpress.org/Plugin_Resources, https://codex.wordpress.org/Plugin_API,
24 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>.) It is a
25 feature of the Accused Instrumentalities that WordPress also contains at least one Player in the form
26 of a runtime player, such that the Application and the Player are provided to the device and executed
27 on the device. (*See*, for example, <https://codex.wordpress.org/CSS>,
28 https://codex.wordpress.org/Using_Javascript; and <https://developer.wordpress.org/rest-api/>.)

1 227. The Accused Instrumentalities feature a system where the Application and Player are
2 provided to the device and executed on the device.

3 228. When the user of the device provides one or more input values associated with an
4 input symbolic name to an input of the defined UI object the device provides the user provided one
5 or more input values and corresponding input symbolic name to the web service. Because the
6 Accused Instrumentalities incorporate a system that includes WordPress, when a user of the device
7 provides one or more input values associated with an input symbolic name, using JSON formatting
8 characteristics, to an input of the defined UI object, the device provides the user provided one or
9 more input values and corresponding input symbolic name, using JSON formatting characteristics, to
10 the web service. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
11 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
12 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API.)

13 229. The Accused Instrumentalities feature a system where the web service utilizes the
14 input symbolic name and the user provided one or more input values for generating one or more
15 output values having an associated output symbolic name. Because of the JSON formatting, the
16 output values having an associated output symbolic name. (*See*, for example,
17 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
18 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
19 https://codex.wordpress.org/Plugin_API.)

20 230. The Accused Instrumentalities feature a system where the Player receives the output
21 symbolic name and corresponding one or more output values and provides instructions for the
22 display of the device to present an output value in the defined UI object. The runtime player within
23 WordPress receives the output name, output value, and provides instructions for a display as shown
24 by the fact that the defined UI object are ultimately rendered. (*See*, for example,
25 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
26 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
27 https://codex.wordpress.org/Plugin_API.)

28

1 231. The presence of the above referenced features is demonstrated, by way of example,
2 by reference to publicly available information. Regarding WordPress, *see, e.g.*,
3 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
4 http://codex.wordpress.org/Template_Hierarchy;
5 http://codex.wordpress.org/Function_Reference/the_title;
6 http://codex.wordpress.org/Function_Reference/the_content;
7 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
8 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
9 http://codex.wordpress.org/Query_Overview. .

10 232. Claim 2 of the '044 patent recites a system for generating code to provide content on
11 a display of a device that includes all the elements of claim 1, additionally where the system stores
12 information in a registry, and wherein the registry includes definitions of input and output related to
13 the web service.

14 233. The Accused Instrumentalities infringe claim 2 of the '044 patent through a
15 combination of features which collectively practice each limitation of claim 2. By way of example,
16 the registry includes definitions of input and output related to a web service as evidenced by
17 WordPress's JSON formatting characteristics of the defined UI objects. (*See*, for example,
18 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
19 <https://developer.wordpress.org/rest-api/reference/posts/> <https://developer.wordpress.org/rest-api/reference/posts/#schema-title> https://codex.wordpress.org/Widgets_API;
20 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
21 https://codex.wordpress.org/WordPress_Widgets.)
22

23 234. Claim 3 of the '044 patent recites a system for generating code to provide content on
24 a display of a device that includes all the elements of claim 1, additionally where the web
25 component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.

26 235. The Accused Instrumentalities infringe claim 3 of the '044 patent through a
27 combination of features which collectively practice each limitation of claim 3. By way of example,
28

1 the Accused Instrumentalities feature web components additionally including web chat, Reuters
2 RSS feed, Calendar image, and map image widgets.

3 236. Claim 4 of the '044 patent recites a system for generating code to provide content on
4 a display of a device that includes all the elements of claim 1, additionally where the UI object is an
5 input field for a chat.

6 237. The Accused Instrumentalities infringe claim 4 of the '044 patent through a
7 combination of features which collectively practice each limitation of claim 4. By way of example,
8 the Accused Instrumentalities additionally feature an UI object that is an input field for a chat.

9 238. Claim 5 of the '044 patent recites a system for generating code to provide content on
10 a display of a device that includes all the elements of claim 1, additionally where the system stores
11 information in a registry, and wherein the registry includes definitions of input and output related to
12 the web service.

13 239. The Accused Instrumentalities infringe claim 5 of the '044 patent through a
14 combination of features which collectively practice each limitation of claim 5. By way of example,
15 the Accused Instrumentalities additionally feature a defined UI object that is an input field for a
16 web service.

17 240. Claim 11 of the '044 patent recites a system for generating code to provide content on
18 a display of a device that includes all the elements of claim 1, additionally where the code is
19 provided over the network.

20 241. The Accused Instrumentalities infringe claim 11 of the '044 patent through a
21 combination of features which collectively practice each limitation of claim 11. WordPress sends all
22 files over a network using a variety of databases in its technology stack including MySQL. These
23 backend capabilities provided the code over a network. By way of example, data from the
24 wp_options table for the website header and from the wp_posts table for the "WordPress Info" web
25 page are extracted directly from the Bitnami WordPress server-side database using MySQL
26 Workbench. The stored data in the wp_options table includes the website's url, the website's title
27 (blogname), the website's tagline (blogdescription), and the active template (style sheet).
28

option_id	option_name	option_value
1	siteurl	http://localhost/wordpress
2	blogname	Express Mobile
3	blogdescription	SELECTED USERS OF WORDPRESS
44	template	the-fundamentals-of-graphic-design

The stored data in the wp_posts table for the “WordPress Info” web page includes information corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages 22%”. Other user selections shown in stored database data below include, for ex-ample, the image filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the image alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for “Heading 3”.

ID	post_content	post_title	post_type
23	<p>WordPress is used by over 14.7% of Alexa Internet's "top 1 million" websites and as of August, 2011, believe it or not, manages 22% of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&nbsp;</p> <p></p> <p>&nbsp;</p> <p>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</p> <p>&nbsp;</p>	WordPress Info	page

WordPress’s HTML, CSS, Java, and JSON coding capabilities further are shown, for example by http://codex.wordpress.org/Templates_ <https://codex.wordpress.org/CSS>, <https://developer.wordpress.org/rest-api/> and https://codex.wordpress.org/Widgets_API).

242. Claim 12 of the ’287 patent recites a system for generating code to provide content on a display of a device that includes all the elements of claim 1, additionally where the defined UI object corresponds to a widget.

1 243. The Accused Instrumentalities infringe claim 12 of the '044 patent through a
2 combination of features which collectively practice each limitation of claim 12. By way of example,
3 the Accused instrumentalities' WordPress includes widgets. *See, e.g.*,
4 <https://codex.wordpress.org/Widgets>.

5 244. The Accused Instrumentalities infringe claim 15 of the '044 patent through a
6 combination of features which collectively practice each limitation of claim 15. By way of example,
7 the Accused Instrumentalities feature a method of displaying content on a display of a device having
8 a Player, in the form of a runtime player and a non-volatile computer memory storing the WordPress
9 MySQL database functionality on the device. The non-volatile computer memory stores symbolic
10 names required for evoking one or more web components each related to a set of inputs and outputs
11 of a web service obtainable over a network, where the symbolic names are character strings that do
12 not contain either a persistent address or pointer to an output value accessible to the web service.
13 The WordPress MySQL database contains symbolic names required for evoking one or more web
14 components each related to a set of inputs and outputs of a web service obtainable over a network as
15 demonstrated by the formatting in conjunction with WordPress's WYSIWYG visual effects editor,
16 widget, and plugin authoring tools.

17 245. Furthermore, each symbolic name has an associated data format class type
18 corresponding to a subclass of UI objects that supports the data format type of the symbolic name,
19 and has a preferred UI object as demonstrated by the presence of JSON formatting in conjunction
20 with WordPress' WYSIWYG visual effects editor and widget authoring tools.

21 246. The computer memory also stores an address of the web service. Because WordPress
22 contains web services, both as a library of Widgets (*see* https://codex.wordpress.org/Widgets_API)
23 and Plug-ins (*see* https://codex.wordpress.org/Plugin_API), it contains the corresponding addresses
24 of the web services. All web services are represented as a wsdl (Web Service Description
25 Language) and wsdl's have URLs to point to the internet location that receives the web service's
26 inputs and returns the web service's outputs (*see* [https://www.soapui.org/soap-and-wsdl/working-](https://www.soapui.org/soap-and-wsdl/working-with-wsdl.html)
27 [with-wsdl.html](https://www.soapui.org/soap-and-wsdl/working-with-wsdl.html)).

28

1 247. The Accused Instrumentalities include defining a UI object for presentation on the
2 display, where the UI object corresponds to a web component included in the computer memory,
3 where the web component is selected from a group consisting of an input of a web service and an
4 output of the web service. WordPress defines a user interface object, *i.e.*, an element/UI component,
5 for presentation on display, where the UI object corresponds to a web component included in the
6 non-volatile computer memory selected from a group consisting of an input of a web service and an
7 output of the web service (as evidenced by JSON data formatting)..

8 248. Each defined UI object is either: 1) selected by a user of an authoring tool; or 2)
9 automatically selected by the system as the preferred UI object corresponding to the symbolic name
10 of the web component selected by the user of the authoring tool. When a Widget is selected in the
11 WordPress Widget selection list (See https://codex.wordpress.org/WordPress_Widgets) steps 1 to 5
12 under “Displaying Widgets“, the widget UI will automatically displayed in the Web Page Sidebar.
13 WordPress contains an authoring tool in the form of the WYSIWYG visual effects editor, widgets,
14 and plug-in authoring tools.

15 249. The Accused Instrumentalities include selecting the symbolic name corresponding to
16 the web component (*i.e.* WordPress Widget or Plug-in) of the defined UI object and associating the
17 selected symbolic name with the defined UI object, where the selected symbolic name is only
18 available to UI objects that support the defined data format associated with that symbolic name.
19 WordPress accesses the non-volatile memory to select the symbolic name corresponding to the web
20 component of the defined UI object (as evidenced by JSON data formatting), associate the selected
21 symbolic name with the defined UI object (the JSON element corresponding to an element), where
22 the selected symbolic name is only available to UI objects that support the defined data format
23 associated with that symbolic name (the element associated with at JSON string).

24 250. The Accused Instrumentalities also include retrieving the information representative
25 of the one or more the UI object settings stored in the database based on the JSON formatting
26 characteristics.

27 251. The Accused Instrumentalities include building an Application consisting of one or
28 more web page views from at least a portion of the database utilizing the Player, where the Player

1 utilizes information stored in the database to generate for the display of at least a portion of the one
2 or more web pages. WordPress builds an Application included in the symbolic name of the defined
3 UI object.

4 252. With the Accused Instrumentalities when the Application and Player are provided to
5 the device and executed on the device when the Application and Player are provided to the device
6 and executed on the device, and when the user of the device provides one or more input values
7 associated with an input symbolic name to an input of the defined UI object, 1) the device provides
8 the user provided one or more input values and corresponding input symbolic name to the web
9 service, 2) the web service utilizes the input symbolic name and the user provided one or more input
10 values for generating one or more output values having an associated output symbolic name, and 3)
11 the Player receives the output symbolic name and corresponding one or more output values and
12 provides instructions for the display of the device to present an output value in the defined UI object.
13 For example, in WordPress, a user of a device provides an input value associated with an input
14 symbolic name to an input of a defined UI object, such as utilizing an element, plug-in, or widget.
15 The element, plug-in, or widget is associated with symbolic name based on their JSON formatting
16 characteristics. 1) The element input value corresponds to the input symbolic name based on its
17 JSON formatting characteristics 2) The element/UI component output value corresponds to the
18 output symbolic name via JSON. 3) The runtime player within WordPress receives the output name,
19 output value, and provides instructions for a display as shown by the fact that the defied UI object is
20 displayed.

21 253. The presence of the above referenced features is demonstrated, by way of example,
22 by reference to publicly available information. Regarding WordPress, see, e.g.,
23 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
24 http://codex.wordpress.org/Template_Hierarchy;
25 http://codex.wordpress.org/Function_Reference/the_title;
26 http://codex.wordpress.org/Function_Reference/the_content;
27 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages1>

28

1 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
2 http://codex.wordpress.org/Query_Overview.

3 254. Claim 16 of the '044 patent recites a method of displaying content on a display that
4 includes all the elements of claim 15, additionally where the method stores information in a registry,
5 and the registry includes definitions of input and output related to the web service.

6 255. The Accused Instrumentalities infringe claim 16 of the '044 patent through a
7 combination of features which collectively practice each limitation of claim 16. By way of example,
8 the Accused Instrumentalities feature storing information in a WordPress registry that includes
9 definitions of input and output related to a web service as evidenced by WordPress's JSON
10 formatting characteristics and the UI object functionality. See, e.g.,
11 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
12 <https://developer.wordpress.org/rest-api/reference/posts/> [https://developer.wordpress.org/rest-](https://developer.wordpress.org/rest-api/reference/posts/#schema-title)
13 [api/reference/posts/#schema-title](https://developer.wordpress.org/rest-api/reference/posts/#schema-title) https://codex.wordpress.org/Widgets_API;
14 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
15 https://codex.wordpress.org/WordPress_Widgets.

16 256. Claim 17 of the '044 patent recites a method of displaying content on a display that
17 includes all the elements of claim 15, additionally where the method additionally where the web
18 component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.

19 257. The Accused Instrumentalities infringe claim 17 of the '044 patent through a
20 combination of features which collectively practice each limitation of claim 17. By way of example,
21 the Accused Instrumentalities feature web components additionally including web chat, Reuters RSS
22 feed, Calendar image, and map image widgets.

23 258. Claim 18 of the '044 patent recites a method of displaying content on a display that
24 includes all the elements of claim 15, additionally where the UI object is an input field for a chat.

25 259. The Accused Instrumentalities infringe claim 18 of the '044 patent through a
26 combination of features which collectively practice each limitation of claim 18. By way of example,
27 the Accused Instrumentalities additionally feature an UI object that is an input field for a chat.
28

1 260. Claim 19 of the '044 patent recites a method of displaying content on a display that
2 includes all the elements of claim 15, additionally where the UI object is an input field for a web
3 service.

4 261. The Accused Instrumentalities infringe claim 19 of the '044 patent through a
5 combination of features which collectively practice each limitation of claim 19. By way of example,
6 the Accused Instrumentalities additionally feature a UI object that is an input field for a web service.

7 262. Claim 25 of the '044 patent recites a method of displaying content on a display that
8 includes all the elements of claim 15, additionally where the method includes providing the code over
9 the network.

10 263. The Accused Instrumentalities infringe claim 25 of the '044 patent through a
11 combination of features which collectively practice each limitation of claim 25. By way of example,
12 WordPress uses a variety of databases including MySQL. These backend capabilities demonstrate
13 that the code is provided to a user over a network. See, e.g., <http://codex.wordpress.org/Templates>.

14 264. Claim 26 of the '044 patent recites a method of displaying content on a display that
15 includes all the elements of claim 15, additionally where the UI object corresponds to a widget.
16 The Accused Instrumentalities infringe claim 26 of the '044 patent through a combination of features
17 which collectively practice each limitation of claim 26. By way of example, the Accused
18 Instrumentalities employ WordPress which includes widgets. See, e.g.,
19 https://codex.wordpress.org/Widgets_API.

20 265. Upon information and belief, these Accused Instrumentalities are used, marketed,
21 provided to, and or used by or for each of Defendant's partners, clients, customers, and/or end users
22 across the country and in this District.

23 266. In particular, Defendant's actions that aid and abet others such as its partners,
24 customers, clients, and/or end users to infringe include advertising and distributing the Accused
25 Instrumentalities and providing instruction materials, training, and services regarding the Accused
26 Instrumentalities. See, e.g., <https://www.rishabhsoft.com/blog/wordpress-or-drupal>. On information
27 and belief, Defendant has engaged in such actions with specific intent to cause infringement or with
28 willful blindness to the resulting infringement because Defendant has had actual knowledge of the

1 '044 patent and knowledge that its acts were inducing infringement of the '044 patent since at least
2 the date Rishabh received notice that such activities infringed the '044 patent.

3 267. Upon information and belief, Defendant is liable as a contributory infringer of the
4 '044 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
5 website authoring tools to be especially made or adapted for use in an infringement of the '044
6 patent. The Accused Instrumentalities are a material component for use in practicing the '044 patent
7 and are specifically made and are not a staple article of commerce suitable for substantial non-
8 infringing use.

9 268. Defendant was made aware of the '044 patent and its infringement thereof at least as
10 early as the filing of this Complaint.

11 269. Since the date of the filing of this Complaint, Defendant's infringement of the '044
12 patent has been willful.

13 270. Within the past six years, Plaintiff has not sold any product nor offered a service
14 within the scope of any claim of the '044 patent. In addition, prior to August 12, 2015, no license to
15 the '044 patent had been granted.

16 271. Plaintiff has been harmed by Defendant's infringing activities.

17 **COUNT V – INFRINGEMENT OF U.S. PATENT NO. 9,063,755**

18 272. The allegations set forth in the foregoing paragraphs 1 through 271 are incorporated
19 into this Fifth Claim for Relief.

20 273. On June 23, 2015, U.S. Patent No. 9,063,755 ("the '755 patent"), entitled "Systems
21 and methods for presenting information on mobile devices," was duly and legally issued by the
22 United States Patent and Trademark Office. A true and correct copy of the '755 patent is attached as
23 Exhibit H.

24 274. The inventions of the '755 patent resolve technical problems related to a system a
25 system for generating code to provide content on a display of a device. The system includes a
26 computer memory and an authoring tool, where the computer memory stores a registry of: a)
27 symbolic names required for evoking one or more web components related to a web service and b)
28 an address of the web service. The authoring tool is configured to: define a (UI) object for

1 presentation on the display, where the defined UI object corresponds to a web component included
2 in the registry selected from a group consisting of an input of the web service and an output of the
3 web service; access the computer memory to select the symbolic name corresponding to the web
4 component of the defined UI object, associate the selected symbolic name with the defined UI
5 object; produce an Application including the selected symbolic name of the defined UI object, where
6 the Application is a device-independent code; and produce a Player, where the Player is a device-
7 dependent code, such that 1) the device provides the user provided one or more input values and
8 corresponding input symbolic name to the web service, 2) the web service utilizes the input symbolic
9 name and the user provided one or more input values for generating one or more output values
10 having an associated output symbolic name, and 3) the Player receives the output symbolic name
11 and corresponding one or more output values and provides instructions. These features are
12 exclusively implemented utilizing computer technology.

13 275. The claims of the '755 patent do not merely recite the performance of some business
14 practice known from the pre-Internet world along with the requirement to perform it on the Internet.
15 Instead, the claims of the '755 patent recite one or more inventive concepts that are rooted in the
16 computerized generation of content on a display of a device, such as a mobile device, and overcome
17 problems specifically arising in the realm of computerized display content generation technologies.

18 276. The claims of the '755 patent recite inventions that are not merely the routine or
19 conventional use of systems and methods for the computerized generation of content on a display of
20 a device. Instead, the inventions feature systems for use with devices and methods of using the
21 systems with authoring tools or Players specific to each device and Applications that are device
22 independent.

23 277. The technology claimed in the '755 patent does not preempt all ways for the
24 computerized generation of code for a display of a device, such as a mobile device, nor preempt the
25 use of all authoring tools or Players for the computerized generation of content on a display of a
26 device, such as a mobile device, nor preempt any other well-known or prior art technology.

27
28

1 278. Accordingly, each claim of the '755 patent recites a combination of elements
2 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an
3 ineligible concept.

4 279. Plaintiff is the assignee and owner of the right, title and interest in and to the '755
5 patent, including the right to assert all causes of action arising under the patents and the right to any
6 remedies for infringement of them.

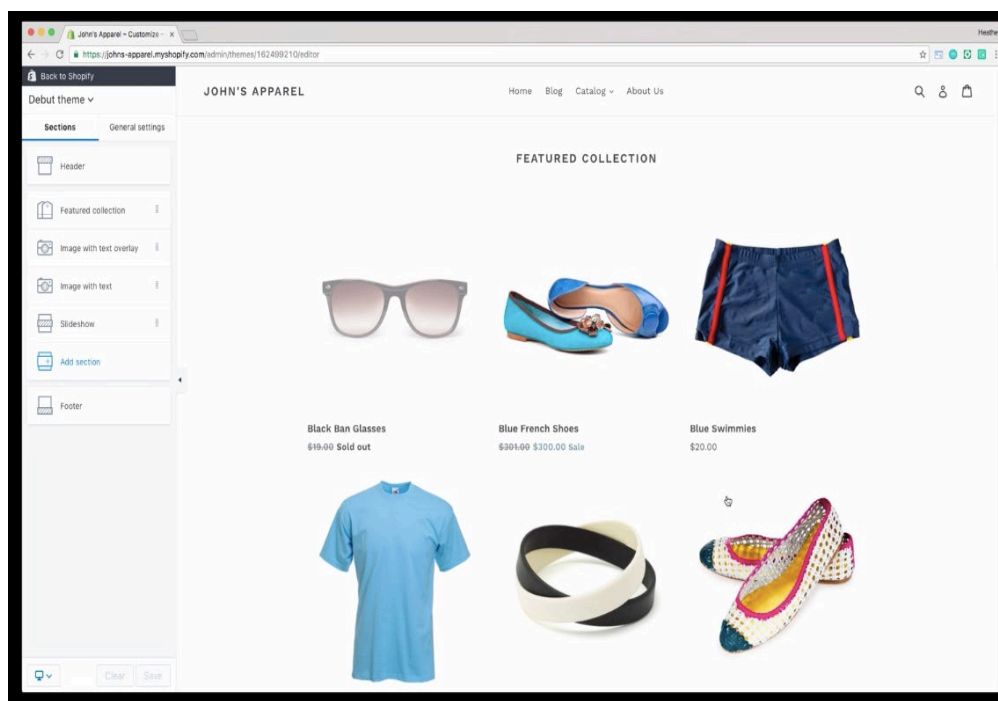
7 280. Upon information and belief, Plaintiffs has and continues to directly infringe at least
8 claims 1 and 15 of the '755 patent by a system a system for generating code to provide content on a
9 display of a device. The system includes a computer memory and an authoring tool, The computer
10 memory stores a registry of: a) symbolic names required for evoking one or more web components
11 each related to a set of inputs and outputs of a web service obtainable over a network, where the
12 symbolic names are character strings that do not contain either a persistent address or pointer to an
13 output value accessible to the web service, where each symbolic name has an associated data format
14 class type corresponding to a subclass of User Interface (UI) objects that support the data format
15 type of the symbolic name, and has a preferred UI object, and b) an address of the web service. The
16 authoring tool is configured to: define a (UI) object for presentation on the display, where the
17 defined UI object corresponds to a web component included in the registry selected from a group
18 consisting of an input of the web service and an output of the web service; access the computer
19 memory to select the symbolic name corresponding to the web component of the defined UI object,
20 associate the selected symbolic name with the defined UI object; produce an Application including
21 the selected symbolic name of the defined UI object, where the Application is a device-independent
22 code; and produce a Player, where the Player is a device-dependent code, such that when the
23 Application and Player are provided to the device and executed on the device, and when the user of
24 the device provides one or more input values associated with an input symbolic name to an input of
25 the defined UI object, 1) the device provides the user provided one or more input values and
26 corresponding input symbolic name to the web service, 2) the web service utilizes the input symbolic
27 name and the user provided one or more input values for generating one or more output values
28 having an associated output symbolic name, 3) the Player receives the output symbolic name and

1 corresponding one or more output values and provides instructions. (The “Accused
2 Instrumentalities”). The Accused Instrumentalities include the Shopify Theme Editor, based on the
3 Ruby on Rails platform, that enables the functionality described above. *See, e.g.*,
4 <https://www.rishabhsoft.com/web-application/ecommerce-development>;
5 [https://medium.com/@chris.chimen/build-shopify-app-with-ruby-on-rails-for-beginer-part-1-
6 40471da7d607](https://medium.com/@chris.chimen/build-shopify-app-with-ruby-on-rails-for-beginer-part-1-40471da7d607) .

7 281. In particular, claim 1 of the ’755 patent recites 1 a system for generating code to
8 provide content on a display of a device, said system comprising: computer memory storing a
9 registry of: a) symbolic names required for evoking one or more web components each related to a
10 set of inputs and outputs of a web service obtainable over a network, where the symbolic names are
11 character strings that do not contain either a persistent address or pointer to an output value
12 accessible to the web service, and b) the address of the web service; an authoring tool configured to:
13 define a user interface (UI) object for presentation on the display, where said UI object corresponds
14 to the web component included in said registry selected from the group consisting of an input of the
15 web service and an output of the web service, access said computer memory to select the symbolic
16 name corresponding to the web component of the defined UI object, associate the selected symbolic
17 name with the defined UI object, produce an Application including the selected symbolic name of
18 the defined UI object, where said Application is a device-independent code, and produce a Player,
19 where said Player is a device-dependent code; such that, when the Application and Player are
20 provided to the device and executed on the device, and when a user of the device provides one or
21 more input values associated with an input symbolic name to an input of defined UI object, 1) the
22 device provides the user provided one or more input values and corresponding input symbolic name
23 to the web service, 2) the web service utilizes the input symbolic name and the user provided one or
24 more input values for generating one or more output values having an associated output symbolic
25 name, 3) said Player receives the output symbolic name and corresponding one or more output
26 values and provides instructions for a display of the device to present an output value in the defined
27 UI object.

28

1 282. The Accused Instrumentalities infringe claim 1 of the '755 patent through a
2 combination of features which collectively practice each limitation of claim 1. By way of example,
3 as shown in the screen shot below, the Accused Instrumentalities feature a method of displaying
4 content of a device.



5
6
7
8
9
10
11
12
13
14
15
16
17 (<https://help.shopify.com/manual/using-themes/change-the-layout>).

18 283. The Accused Instrumentalities feature a registry of one or more web components
19 related to inputs and outputs of a web service obtainable over a network. The registry is provided by
20 the Shopify servers and their MySQL database. By way of example, the Accused Instrumentalities'
21 Ruby on Rails platform utilizes JSON strings extensively as part of its API, which necessarily
22 require servers and databases. (See, <https://developer.Ruby on Rails.org/rest-api/>.)

23 284. Each web component includes a plurality of symbolic names of inputs and outputs
24 associated with each web service.

25 285. Furthermore, the registry required for evoking one or more web components each
26 related to a set of inputs and outputs of a web service obtainable over a network, where the symbolic
27 names are character strings that do not contain either a persistent address or pointer to an output
28 value accessible to the web service. In particular, the Accused Instrumentalities' MySQL database

1 contains symbolic names required for evoking one or more web components each related to a set of
2 inputs and outputs of a web service obtainable over a network by the formatting of the symbolic
3 names in conjunction with the Shopify Theme Editor, widget, and plugin authoring tools.

4 286. The computer memory also stores b) an address of the web service. Because the
5 Shopify Theme Editor contains web services, it contains the corresponding addresses for the web
6 services.

7 287. The Accused Instrumentalities feature an authoring tool in the form of the Shopify
8 Theme Editor, widget, and plug-in authoring tools.

9 288. The authoring tool is configured to define a UI object for presentation on the display,
10 where the defined UI object corresponds to a web component included in the registry selected from a
11 group consisting of an input of the web service and an output of the web service. The Shopify
12 Theme Editor's drag and drop component capabilities define the presence of a (UI) object for
13 presentation on a display, where the defined UI object corresponds to a web component included in
14 the registry (Shopify Server and mySQLdatabase) selected from a group consisting of an input of the
15 web service and an output of the web service.

16 289. The Accused Instrumentalities' authoring tool is configured to access the computer
17 memory to select the symbolic name corresponding to the web component of the defined UI object
18 by a JSON formatted element.

19 290. The Accused Instrumentalities' authoring tool is also configured to associate the
20 selected symbolic name with the defined UI object, i.e., the JSON formatted element, where the
21 selected symbolic name is only available to UI objects that support the defined data format
22 associated with the element associated with that symbolic name, i.e., JSON string. JSON names are
23 strings that only represent the symbolic names that are bound both to a web service input and/or
24 output and to a UI object. All JSON names in the name/value pairs are character strings. When the
25 Shopify Theme Editor makes a UI element request a JSON request is sent to the Server and a JSON
26 data response is provided to the Interface.

27 291. The Accused Instrumentalities authoring tool is configured to produce an Application
28 including the selected symbolic name of the defined UI object, where the Application is a device-

1 independent. The Accused Instrumentalities' Application is comprised of the MySQL database,
2 including all the user selectable settings, as augmented by the web component definition. The
3 application, because it contains user selected settings, and is represented by Boolean, numbers and
4 String primitives, is device independent, and stored in the MySQL database for each application.

5 292. The Accused Instrumentalities authoring tool is further configured to produce a
6 Player, where the Player is a device-dependent code. The Shopify Theme Editor contains a Player in
7 the form of a run time player. The Accused Instrumentality produces a device dependent file, which
8 is wrapped inside the run time file, i.e., *i.e.*, a file, including a run time engine, that is downloaded or
9 created when a browser is pointed to a web page or website,. In order for a site to display on
10 different devices through a browser or through responsive capabilities, there is device dependent
11 code. *See, e.g., X Commerce, Inc. v Express Mobile, Inc.*, Case No 17-cv-02605-RS, NDCA, DKT
12 79-5.

13 293. The Accused Instrumentalities feature a system where the Application and Player are
14 provided to the device and executed on the device and when the user of the device provides one or
15 more input values associated with an input symbolic name to an input of the defined UI object.
16 Because the Accused Instrumentalities incorporate a system that includes Ruby on Rails, when a
17 user of the device provides one or more input values associated with an input symbolic name, using
18 JSON formatting characteristics, to an input of the defined UI object, the device provides the user
19 provided one or more input values and corresponding input symbolic name, using JSON formatting
20 characteristics, to the web service.

21 294. The Accused Instrumentalities feature a system where the device provides the user
22 provided one or more input values and corresponding input symbolic name to the web service.
23 Because the Accused Instrumentalities incorporate a system that includes the Ruby on Rails
24 platform, the web service utilizes the input symbolic name and the user provided one or more input
25 values for generating one or more output values having an associated output symbolic name. The
26 defined UI object output value corresponds to the output symbolic name based on its JSON
27 formatting characteristics.

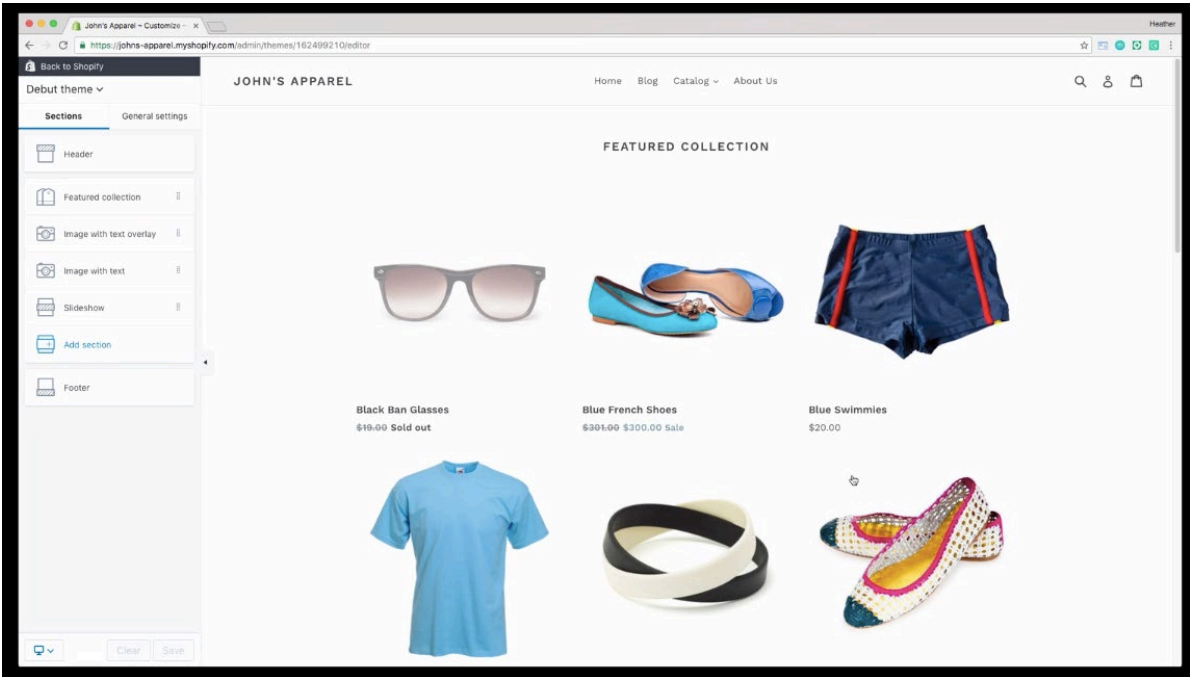
28

1 295. The Accused Instrumentalities feature a system where the web service utilizes the
2 input symbolic name and the user provided one or more input values for generating one or more
3 output values having an associated output symbolic name. Because of the JSON formatting, the
4 output values having an associated output symbolic name.

5 296. The Accused Instrumentalities feature a system where the Player receives the output
6 symbolic name and corresponding one or more output values and provides instructions for the
7 display of the device to present an output value in the defined UI object. The runtime player within
8 Ruby on Rails receives the output name, output value, and provides instructions for a display as
9 shown by the fact that the defined UI object are ultimately rendered.

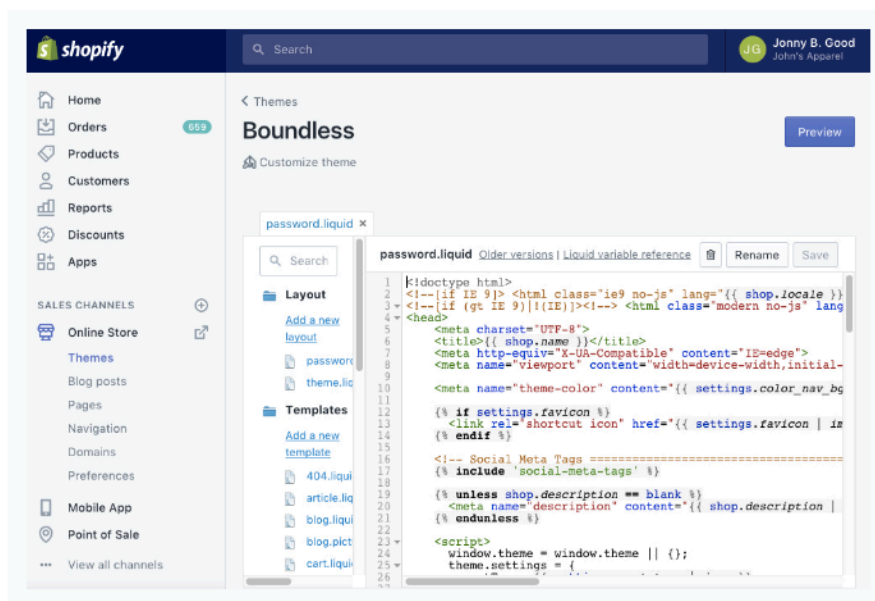
10 297. The presence of the above referenced features is demonstrated, by way of example,
11 by reference to publicly available information, including <https://www.shopify.com/>;
12 <https://www.shopify.com/website/hosting>; [https://help.shopify.com/manual/using-themes/change-](https://help.shopify.com/manual/using-themes/change-the-layout)
13 [the-layout](https://help.shopify.com/manual/using-themes/change-the-layout/using-theme-presets); <https://help.shopify.com/manual/using-themes/change-the-layout/using-theme-presets>
14 <https://help.shopify.com/manual/using-themes/troubleshooting/fix-64-kilobyte-limit-errors>;
15 <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>; [https://www.slideshare.net/jduff/how-](https://www.slideshare.net/jduff/how-shopify-scales-rails-20443485)
16 [shopify-scales-rails-20443485](http://techstacks.io/shopify); <http://techstacks.io/shopify>; [https://help.shopify.com/manual/intro-](https://help.shopify.com/manual/intro-to-shopify/pricing-plans/plan-features)
17 [to-shopify/pricing-plans/plan-features](https://help.shopify.com/api/reference); <https://help.shopify.com/api/reference>;
18 <https://help.shopify.com/api/storefront-api> [https://help.shopify.com/api/sdks/shopify-](https://help.shopify.com/api/sdks/shopify-apps/modifying-online-store/use-javascript-responsibly)
19 [apps/modifying-online-store/use-javascript-responsibly](https://help.shopify.com/api/reference/product);
20 <https://help.shopify.com/api/reference/product>; [https://dev.mysql.com/doc/refman/5.7/en/what-is-](https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html)
21 [mysql.html](http://techstacks.io/shopify); <http://techstacks.io/shopify>; <https://help.shopify.com/api/reference>;
22 <https://help.shopify.com/manual/apps>; <https://api.rubyonrails.org/>;
23 https://guides.rubyonrails.org/rails_application_templates.html;
24 <https://teamtreehouse.com/community/ruby-on-rails-with-html-templates-2>;
25 <https://guides.railsgirls.com/design>; <https://guides.rubyonrails.org/plugins.html>;
26 <https://www.rubydoc.info/gems/dashing-rails>; [https://github.com/Shopify/dashing/wiki/Additional-](https://github.com/Shopify/dashing/wiki/Additional-Widgets)
27 [Widgets](https://help.shopify.com/themes/customization/communication/add-contact-form); <https://help.shopify.com/themes/customization/communication/add-contact-form>; and
28 <https://help.shopify.com/manual/using-themes/change-the-layout/add-video>.

1 298. The Accused Instrumentalities infringe claim 12 of the '755 patent through a
2 combination of features which collectively practice each limitation of claim 12. By way of example,
3 the method is for displaying content on a display of a device as shown in the screen shot below.
4



5
6
7
8
9
10
11
12
13
14
15
16 <https://help.shopify.com/manual/using-themes/change-the-layout>).
17
18
19
20
21
22
23
24
25
26
27
28

Edit HTML/CSS page



The **Edit HTML/CSS** page can be accessed through the **Themes** page of your Shopify Admin. It allows you to see all of your theme templates and assets, and directly make changes to them.

(<https://help.shopify.com/themes/development/getting-started/choosing-an-editor>).

299. The Accused Instrumentalities feature a registry of one or more web components, inputs and outputs of a web service obtainable over a network and an output of the web service by JSON data formatting. JSON names are strings that only represent the symbolic names that are bound both to a web service input and/or output and to a UI object. All JSON names in the name/value pairs are character strings. When the Ruby on Rails interface makes a UI element request a JSON request is sent to the Server and a JSON data response is provided to the Interface.

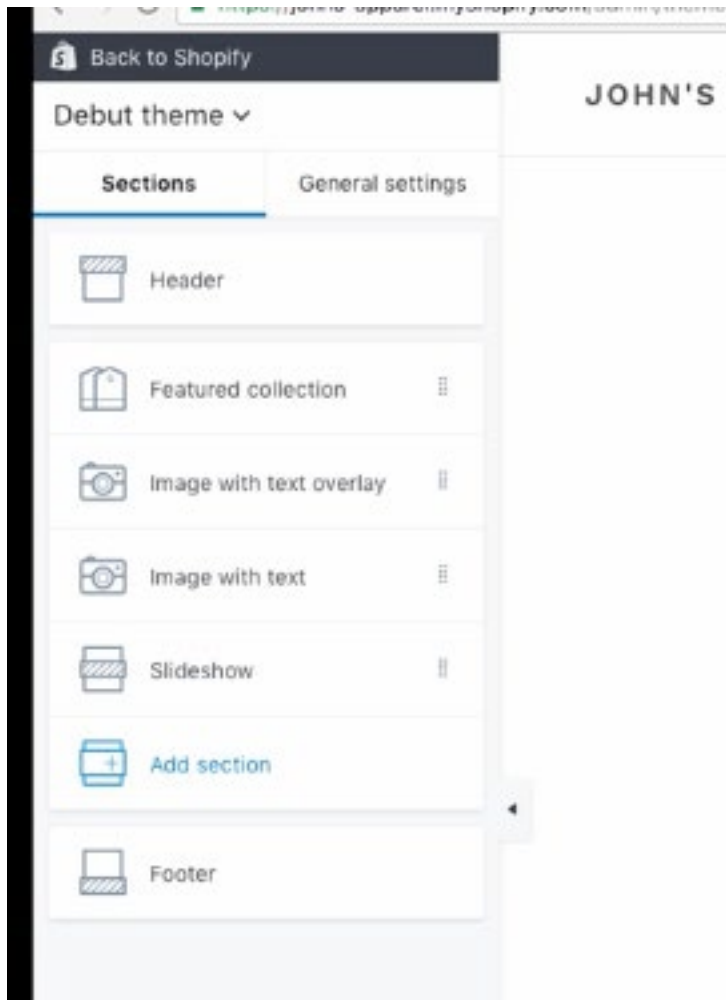
300. Each web component includes a plurality of symbolic names of inputs and outputs associated with each web service. The plurality of symbolic names of inputs and outputs associated with each web service is a feature of their JSON formatting characteristics. Each symbolic name has an associated data format class type corresponding to a subclass of UI objects that supports the data format type of the symbolic name, and has a preferred UI object as demonstrated by the presence of JSON formatting in conjunction with the Shopify Theme Editor and widget capabilities.

1 301. The registry includes: a) symbolic names required for evoking one or more web
2 components each related to a set of inputs and outputs of a web service obtainable over a network,
3 where the symbolic names are character strings that do not contain either a persistent address or
4 pointer to an output value accessible to the web service. The registry and Ruby on Rails MySQL
5 database contain symbolic names required for evoking one or more web components each related to
6 a set of inputs and outputs of a web service obtainable over a network as demonstrated by the
7 formatting in conjunction with the Shopify Theme Editor, widget, and plugin authoring tools. JSON
8 names are strings that only represent the symbolic names that are bound both to a web service input
9 and/or output and to a UI object. All JSON names in the name/value pairs are character strings.
10 When the Ruby on Rails interface makes a UI element request a JSON request is sent to the Server
11 and a JSON data response is provided to the Interface.

12 302. The registry also includes b) an address of the web service. Because the Shopify
13 Theme Editor contains web services, it contains the corresponding web addresses.

14 303. Accused Instrumentalities define a user interface (UI) object for presentation on the
15 display, where said UI object corresponds to a web component included in said registry selected
16 from the group consisting of an input of the web service and an output of the web service. The
17 Accused Instrumentalities contain an authoring tool in the form of the Shopify Theme Editor shown
18 in the screen shot below.

19
20
21
22
23
24
25
26
27
28



(<https://help.shopify.com/manual/using-themes/change-the-layout>).

The ShopifyTheme Editor's drag and drop component capabilities define the presence of a (UI) object for presentation on a display, where the defined UI object corresponds to a web component included in the registry selected from a group consisting of an input of the web service and an output of the web service.

304. The Accused Instrumentalities include selecting the symbolic name from the web component (i.e Shopify Theme EditorWidget or Plug-in) corresponding to the defined UI object, where the selected symbolic name has an associated data format class type corresponding to a subclass of UI objects that support the data format type of the symbolic name and has the preferred UI object. The Shopify Theme Editor accesses its memory to select the symbolic name

1 corresponding to the web component of the defined UI object (as evidenced by JSON data
2 formatting), associate the selected symbolic name with the defined UI object (the JSON element
3 corresponding to an element), where the selected symbolic name is only available to UI objects that
4 support the defined data format associated with that symbolic name (the element associated with at
5 JSON string). Additionally, the preferred UI object is the selected UI object. JSON names are
6 strings that only represent the symbolic names that are bound both to a web service input and/or
7 output and to a UI object. All JSON names in the name/value pairs are character strings. When the
8 Ruby on Rails interface makes a UI element request a JSON request is sent to the Server and a JSON
9 data response is provided to the Interface.

10 305. The Accused Instrumentalities include associating the selected symbolic name with
11 the defined UI object.

12 306. The Accused Instrumentalities include producing an Application including the
13 selected symbolic name of the defined UI object, where the Application is a device-independent
14 code. The Application is comprised of the MySQL database that includes all the user selectable
15 settings, augmented by the web component definition. The application, because it contains user
16 selected settings, and is represented by Boolean, numbers and String primitives, is device
17 independent, and stored in a database for each application.

18 307. The Accused Instrumentalities also include producing a Player, where the Player is a
19 device-dependent code. The Shopify Theme Editor contains a Player in the form of a runtime
20 player.

21 308. The Accused Instrumentalities feature a system where the Application and Player are
22 provided to the device and executed on the device and when the user of the device provides one or
23 more input values associated with an input symbolic name to an input of the defined UI object.
24 Because the Accused Instrumentalities incorporate a system that includes the Shopify Theme Editor,
25 when a user of the device provides one or more input values associated with an input symbolic
26 name, using JSON formatting characteristics, to an input of the defined UI object, the device
27 provides the user provided one or more input values and corresponding input symbolic name, using
28 JSON formatting characteristics, to the web service.

1 309. The Accused Instrumentalities feature a system where the device provides the user
2 provided one or more input values and corresponding input symbolic name to the web service.
3 Because the Accused Instrumentalities incorporate a system that includes Ruby on Rails, the web
4 service utilizes the input symbolic name and the user provided one or more input values for
5 generating one or more output values having an associated output symbolic name. The defined UI
6 object output value corresponds to the output symbolic name based on its JSON formatting
7 characteristics.

8 310. The Accused Instrumentalities feature a system where the web service utilizes the
9 input symbolic name and the user provided one or more input values for generating one or more
10 output values having an associated output symbolic name. Because of the JSON formatting, the
11 output values having an associated output symbolic name.

12 311. The Accused Instrumentalities feature a system where the Player receives the output
13 symbolic name and corresponding one or more output values and provides instructions for the
14 display of the device to present an output value in the defined UI object. The runtime player within
15 Ruby on Rails receives the output name, output value, and provides instructions for a display as
16 shown by the fact that the defined UI object are ultimately rendered.

17 312. The presence of the above referenced features is demonstrated, by way of example,
18 by reference to publicly available information includig. <https://www.shopify.com/>;
19 <https://www.shopify.com/website/hosting>; [https://help.shopify.com/themes/development/getting-](https://help.shopify.com/themes/development/getting-started/choosing-an-editor)
20 [started/choosing-an-editor](https://help.shopify.com/manual/using-themes/change-the-layout); <https://help.shopify.com/manual/using-themes/change-the-layout>;
21 <https://www.shopify.com/online>; [https://help.shopify.com/api/sdks/shopify-apps/apps-on-shopify-](https://help.shopify.com/api/sdks/shopify-apps/apps-on-shopify-mobile)
22 [mobile](https://developers.shopify.com/mobile-buy-sdk); <https://developers.shopify.com/mobile-buy-sdk>; [https://help.shopify.com/manual/using-](https://help.shopify.com/manual/using-themes/change-the-layout/using-theme-presets)
23 [themes/change-the-layout/using-theme-presets](https://help.shopify.com/manual/using-themes/change-the-layout/using-theme-presets) [https://help.shopify.com/manual/using-](https://help.shopify.com/manual/using-themes/troubleshooting/fix-64-kilobyte-limit-errors)
24 [themes/troubleshooting/fix-64-kilobyte-limit-errors](https://help.shopify.com/manual/using-themes/troubleshooting/fix-64-kilobyte-limit-errors); [https://dev.mysql.com/doc/refman/5.7/en/what-](https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html)
25 [is-mysql.html](https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html); <https://www.slideshare.net/jduff/how-shopify-scales-rails-20443485>;
26 <http://techstacks.io/shopify>; [https://help.shopify.com/manual/intro-to-shopify/pricing-plans/plan-](https://help.shopify.com/manual/intro-to-shopify/pricing-plans/plan-features)
27 [features](https://help.shopify.com/manual/intro-to-shopify/pricing-plans/plan-features); <https://help.shopify.com/api/reference>; <https://help.shopify.com/api/storefront-api>
28 <https://help.shopify.com/api/sdks/shopify-apps/modifying-online-store/use-javascript-responsibly>;

1 <https://help.shopify.com/api/reference/product>; [https://dev.mysql.com/doc/refman/5.7/en/what-is-](https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html)
2 [mysql.html](https://techstacks.io/shopify); <http://techstacks.io/shopify>; <https://help.shopify.com/api/reference>;
3 <https://help.shopify.com/manual/apps>; <https://api.rubyonrails.org/>;
4 https://guides.rubyonrails.org/rails_application_templates.html;
5 <https://teamtreehouse.com/community/ruby-on-rails-with-html-templates-2>;
6 <https://guides.railsgirls.com/design>; <https://guides.rubyonrails.org/plugins.html>;
7 <https://www.rubydoc.info/gems/dashing-rails>; [https://github.com/Shopify/dashing/wiki/Additional-](https://github.com/Shopify/dashing/wiki/Additional-Widgets)
8 [Widgets](https://help.shopify.com/themes/customization/communication/add-contact-form); <https://help.shopify.com/themes/customization/communication/add-contact-form>; and
9 <https://help.shopify.com/manual/using-themes/change-the-layout/add-video>.

10 314. Upon information and belief, these Accused Instrumentalities are used, marketed,
11 provided to, and or used by or for each of Defendant's partners, clients, customers, and/or end users
12 across the country and in this District.

13 315. In particular, Defendant's actions that aid and abet others such as its partners,
14 customers, clients, and/or end users to infringe include advertising and distributing the Accused
15 Instrumentalities and providing instruction materials, training, and services regarding the Accused
16 Instrumentalities. *See, e.g.*, [https://www.rishabhsoft.com/blog/office-365-10-new-countries-aws-](https://www.rishabhsoft.com/blog/office-365-10-new-countries-aws-slashes-storage-rates-shopify-frenzy)
17 [slashes-storage-rates-shopify-frenzy](https://www.rishabhsoft.com/blog/office-365-10-new-countries-aws-slashes-storage-rates-shopify-frenzy). On information and belief, Defendant has engaged in such
18 actions with specific intent to cause infringement or with willful blindness to the resulting
19 infringement because Defendant has had actual knowledge of the '755 patent and knowledge that its
20 acts were inducing infringement of the '755 patent since at least the date Rishabh received notice
21 that such activities infringed the '755 patent.

22 316. Upon information and belief, Defendant is liable as a contributory infringer of the
23 '755 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
24 website authoring tools to be especially made or adapted for use in an infringement of the '755
25 patent. The Accused Instrumentalities are a material component for use in practicing the '755 patent
26 and are specifically made and are not a staple article of commerce suitable for substantial non-
27 infringing use.

28

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

Dated: June 13, 2019

By: /s/ Jeffrey Francis Craft
Jeffrey Francis Craft (SBN 147186)
jcraft@devlinlawfirm.com
DEVLIN LAW FIRM LLC
1731 Fox Springs Circle,
Newbury Park, CA 91320

Attorneys for Plaintiff Express Mobile, Inc.