

1 Seth W. Wiener (SBN 203747)
seth@sethwienerlaw.com
2 LAW OFFICES OF SETH WIENER
9107 Wilshire Blvd., Suite 450
3 Beverly Hills, CA 90210
Telephone: (925) 487-5607

4 Jeffrey Francis Craft (SBN 147186)
5 jcraft@devlinlawfirm.com
DEVLIN LAW FIRM LLC
6 1731 Fox Springs Circle,
Newbury Park, CA 91320

7 Timothy Devlin (*pro hac vice* to be filed)
8 tdevlin@devlinlawfirm.com
Robert Kiddie (*pro hac vice* to be filed)
9 rkiddie@devlinlawfirm.com
DEVLIN LAW FIRM LLC
10 1526 Gilpin Avenue
Wilmington, DE 19806
11 Telephone: (302) 449-9010
Facsimile: (302) 353-4251

12 Attorneys for Express Mobile, Inc.

13
14 UNITED STATES DISTRICT COURT
15 CENTRAL DISTRICT OF CALIFORNIA
16 WESTERN DIVISION
17

18 EXPRESS MOBILE, INC.,

19 Plaintiff,

20 vs.

21
22 ADVANTAGE AMP, INC. d/b/a AMP
23 AGENCY

24 Defendant.
25
26
27
28

) Case No.: 2:19-cv-5155

) **COMPLAINT FOR PATENT
INFRINGEMENT**

) DEMAND FOR JURY TRIAL

1 Plaintiff Express Mobile, Inc. (“Express Mobile” or “Plaintiff”), for its
2 Complaint against Defendant Advantage AMP, Inc. d/b/a AMP Agency, (“AMP” or
3 “Defendant”) alleges the following:

4 **NATURE OF THE ACTION**

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

7 **THE PARTIES**

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

10 3. Upon information and belief, AMP is a corporation organized and
11 existing under the laws of Delaware, with a place of business at 6080 Center Drive,
12 Suite 200, Los Angeles, CA 90045 and can be served through its registered agent,
13 Corporation Service Company, 251 Little Falls Dr., Wilmington, DE 19808.

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

19 **JURISDICTION AND VENUE**

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

22 6. This Court has subject matter jurisdiction under 28 U.S.C. §§ 1331 and
23 1338(a).

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

27 8. On information and belief, Defendant is subject to this Court’s general
28 and specific personal jurisdiction because Defendant has sufficient minimum contacts

1 within the State of California and this District, pursuant to due process and/or the
2 California Long Arm Statute because Defendant purposefully availed itself of the
3 privileges of conducting business in the State of California and in this District,
4 because Defendant regularly conducts and solicits business within the State of
5 California and within this District, and because Plaintiff's causes of action arise
6 directly from each of Defendant's business contacts and other activities in the State of
7 California and this District.

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

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

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

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

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

27 13. The claims of the '397 patent recite an invention that is not merely the
28 routine or conventional use of website creation systems and methods. Instead, the

1 invention describes a browser-based website creation system and method in which the
2 user-selected settings representing website elements are stored in a database, and in
3 which said stored information is retrieved to generate said website.

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

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

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

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

1 18. In C.A. Nos. 3:18-cv-04679 and 3:18-04688, both of which were filed in
2 the Northern District of California, the respective defendant in each of those actions
3 brought a Motion to Dismiss asserting that the '397 patent, along with U.S. Patent No.
4 7,594,168 (asserted in Count II below) were invalid as claiming abstract subject
5 matter under 35 U.S.C. § 101. The § 101 briefing in each of those cases is
6 incorporated by reference into this Complaint.

7 19. After consideration of the respective pleadings and oral argument, Judge
8 Richard Seeborg issued orders denying each respective motion to dismiss drawing a
9 comparison between the asserted Express Mobile patents with those patents asserted
10 in *Enfish, LLC v. Microsoft Corp.*, 822 F.3d 1327 (Fed. Cir. 2016). (C.A. 3:18 -04679
11 Dkt. 45, attached hereto as Exhibit C, and C.A. 3:18-04688 Dkt. 40, attached hereto as
12 Exhibit D.)

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

16 21. Upon information and belief, Defendant has and continues to directly
17 infringe at least claims 1-6, 8-11, 14-15, 17, 20, 24-25, 35, and 37 of the '397 patent
18 by using a browser-based website and/or web page authoring tool in which the user-
19 selected settings representing website elements are stored in a database, and in which
20 said stored information is retrieved to generate said website (the "Accused
21 Instrumentalities"). The Accused Instrumentalities include but are not limited to the
22 website building tools used and/or provided by Defendant, such as, for example
23 Drupal, Magento and/or Wordpress. *See, e.g.*,
24 <https://www.ampagency.com/livenation-insomniac-entertainment-media-case-study>;
25 <https://www.ampagency.com/sena-cases-accessories-case-study>; and
26 <https://www.ampagency.com/mobile-web-development-and-web-design-agency>.

27 22. On information and belief, Defendant is a for-profit organization with
28 revenues of approximately \$70 million U.S.D. per year. Moreover, Defendant, its

1 employees and/or agents utilize the Accused Instrumentalities in the building and/or
2 hosting of websites for Defendant's customers, leading to direct or indirect revenues
3 and profit. As one example of indirect profit, entities such as Defendant will
4 frequently offer website building and/or hosting services at reduced pricing as an
5 inducement to attract customers, who then purchase additional products or services.
6 On information and belief, without the availability of infringing tools such as the
7 Accused Instrumentalities, Defendant would be at a disadvantage in the marketplace
8 and would generate less revenue overall.

9 23. In particular, claim 1 of the '397 patent generally recites a method
10 enabling production of websites on and for computers with browsers and virtual
11 machines, by presenting, through a browser, a selectable settings menu describing
12 elements, such setting(s) corresponding to commands to the virtual machine;
13 generating a display in accordance with selected settings; storing information
14 regarding selected settings in a database; generating a website at least in part by
15 retrieving said information; and building web page(s) to generate said website and a
16 run time file, where the run time file uses the stored information to generate virtual
17 machine commands for the display of at least a portion of web page(s).

18 24. The Accused Instrumentalities infringe claim 1 of the '397 patent through
19 a combination of features which collectively practice each limitation of claim 1. By
20 way of example, modern internet browsers such as Microsoft Internet Explorer,
21 Mozilla's Firefox, Apple Safari, Google Chrome, and Opera include virtual machines
22 within the meaning of the '397 patent. (*See, e.g.*,
23 <http://developer.telerik.com/featured/a-guide-to-javascript-engines-for-idiots/>;
24 <http://dictionary.reference.com/browse/virtual+machine?s=t>). The Accused
25 Instrumentalities support the use of the latest versions of Internet Explorer 11 or later,
26 Microsoft Edge, latest-1, Firefox latest, latest-1, Chrome latest, latest-1, Safari latest,
27 latest-1 (Mac OS), Safari Mobile for iPad 2, iPad Mini, iPad with Retina Display (iOS
28 7 or later), for desktop site, Safari Mobile for iPhone 4 or later; iOS 7 or later, for

1 mobile site, Chrome for mobile latest-1 (Android 4 or later) for mobile site, where
2 *latest-1* means one major version earlier than the latest released version. (See, e.g.,
3 <https://www.drupal.org/docs/8/system-requirements/browser->;
4 [http://devdocs.magento.com/guides/v2.0/install-gde/system-](http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html)
5 [requirements_browsers.html](http://themeforest.net/category/wordpress); <http://themeforest.net/category/wordpress>.) All of these
6 browsers rely on browser engines comprising virtual machines to interpret and
7 execute JavaScript and HTML to render web pages on a computer.

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

24 26. Data is stored in a database, including information corresponding to user
25 selected settings such as, for example, the selections of text color. Other user
26 selections are also stored including, for example, the layout, image filenames,
27 thumbnails, and paragraph margin settings for defining the alignment of an image
28 location. The Accused Instrumentalities build one or more web pages to generate a

1 website from at least a portion of a database and at least one run time file, where at
2 least one run time file utilizes information stored in said database to generate virtual
3 machine commands for the display of at least a portion of said one or more web pages.

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

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

18 29. The presence of the above referenced elements are demonstrated, by way
19 of example, by reference to publicly available information. Regarding Drupal, *see,*
20 *e.g.,* <https://www.drupal.org/home>; <https://www.drupal.org/docs/8/system-requirements/browser-requirements>; <https://www.drupal.org/project/ckeditor>;
21 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
22 [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)
23 [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);
24 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
25 <https://www.drupal.org/docs/8/system-requirements/web-server>;
26 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
27 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
28

1 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
2 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>.
3 Regarding Magento, *see, e.g.*, [http://docs.magento.com/m1/ee/user_guide/system-](http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html)
4 [operations/browser-capabilities-detection.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html);
5 [http://docs.magento.com/m1/ee/user_guide/system-](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html)
6 [management.html](http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html); [http://docs.magento.com/m1/ce/user_guide/cms/magento-](http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html)
7 [cms.html](http://docs.magento.com/m1/ce/user_guide/cms/page-create.html); http://docs.magento.com/m1/ce/user_guide/cms/page-create.html;
8 http://docs.magento.com/m1/ce/user_guide/cms/editor.html;
9 http://docs.magento.com/m1/ce/user_guide/cms/links.html;
10 http://docs.magento.com/m1/ce/user_guide/cms/editor-insert-image.html;
11 http://docs.magento.com/m1/ce/user_guide/cms/editor-add-widget.html;
12 http://docs.magento.com/m1/ce/user_guide/design/page-layout.html;
13 http://docs.magento.com/m1/ce/user_guide/design/layout-updates.html;
14 http://docs.magento.com/m1/ee/user_guide/store-operations/stores-multiple.html;
15 http://docs.magento.com/m1/ee/user_guide/store-operations/store-hierarchy.html;
16 [http://docs.magento.com/m1/ee/user_guide/system-operations/index-](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html)
17 [management.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html). Regarding Wordpress, *see, e.g.*,
18 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
19 http://codex.wordpress.org/Template_Hierarchy;
20 http://codex.wordpress.org/Function_Reference/the_title;
21 http://codex.wordpress.org/Function_Reference/the_content;
22 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
23 <http://codex.wordpress.org/Templates>;
24 http://codex.wordpress.org/Template_Tags/get_the_title; and
25 http://codex.wordpress.org/Query_Overview.

26 30. Claim 2 of the '397 patent generally recites an apparatus for producing
27 websites on and for computers having a browser and a virtual machine, said apparatus
28 comprising an interface to present a settings menu which describes elements, said

1 panel presented through a browser, where the selectable setting(s) corresponds to
2 commands to the virtual machine; a browser to generate a display in accordance with
3 selected setting(s); a database for storing information regarding selected settings; and
4 a build tool having run time file(s) for generating web page(s) and using stored
5 information to generate commands to the virtual machine for generating at least a
6 portion of web page(s).

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

25 32. By way of further example, the Accused Instrumentalities enable users to
26 produce websites through browsers on users' computers via interaction with an
27 Internet server. For example, in order to add a new page to a user's website, the user
28 logs in and then a server of the Accused Instrumentalities initiates presentation to the

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

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

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

28

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

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

8 37. The presence of the above referenced elements are demonstrated, by way
9 of example, by reference to publicly available information. Regarding Drupal, *see,*
10 *e.g.*, <https://www.drupal.org/home>; <https://www.drupal.org/docs/8/system-requirements/browser-requirements>; <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-
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://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
15 <https://www.drupal.org/docs/8/system-requirements/web-server>;
16 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
17 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
18 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
19 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>.
20 Regarding Magento, *see, e.g.*, [http://docs.magento.com/m1/ee/user_guide/system-
21 operations/browser-capabilities-detection.html](http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html);
22 [http://docs.magento.com/m1/ee/user_guide/system-operations/index-
23 management.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html); [http://docs.magento.com/m1/ce/user_guide/cms/magento-
24 cms.html](http://docs.magento.com/m1/ce/user_guide/cms/magento-cms.html); 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

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

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

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

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

1 <https://code.tutsplus.com/>;

2 [https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)
3 [option-array-structure.](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)

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

7 42. The Accused Instrumentalities infringe claim 4 of the '397 patent through
8 a combination of features that practice the limitations of Claim 4. *See, e.g.,*
9 [https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-](https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions)
10 [definitions;](https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions)

11 <http://devdocs.magento.com/guides/mlx/api/rest/Resources/Products/products.html>;
12 [https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)
13 [option-array-structure.](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)

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

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

19 45. By way of example, the Accused Instrumentalities include various
20 multimedia objects selected from a group contained within a WYSIWYG Editor.
21 Examples include color, font, an image, a video, a text area and a URL as they appear
22 in the WYSIWYG Editor. The multimedia objects created in the WYSIWYG editor
23 are stored in the database and appear as HTML scripted text in the database. Text and
24 vector objects can be selected and colored by selecting them or “click and dragging”
25 over them in the WYSIWYG editor. A color may also be selected from the color
26 dropdowns on the control bar of the Editor. This color is saved to the database; as part
27 of the HTML of the description record. Moreover, text objects may be assigned a font
28 by making such a selection or “click and dragging” over them in the WYSIWYG

1 editor. A font can then be selected from the font dropdown on the control bar of the
2 Editor. This font selection is thereafter saved to the database as part of the HTML of
3 the description record. Selecting the Image button in the WYSIWYG editor opens a
4 tabbed panel where the user designates source, title, format, size, etc. The image file
5 is uploaded to the server and the file's location and style are saved and posted to the
6 database as part of the HTML of the description record. Furthermore, videos are
7 created by clicking on the Media module, which opens a tabbed panel where the user
8 designates URL, format, size, etc. The video's URL and style elements are saved to
9 the database as part of the HTML of the description record. A text area may also be
10 selected for creation by clicking in the frame of the WYSIWYG Editor and typing.
11 The text and its style are saved to the database as part of the HTML of the description
12 record. After entering text into the WYSIWYG editor's text area, a URL assigned by
13 clicking and dragging over the text object you wish to link, and then selecting the
14 "chain" link button from the control bar; which opens a tabbed panel where the user
15 can designate the URL, target, etc. The text and its style are saved to the database as
16 part of the HTML of the description record.

17 46. The presence of the above referenced elements are demonstrated, by way
18 of example, by reference to publicly available information. *See, e.g.*,
19 <https://www.drupal.org/project/ckeditor>;
20 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
21 [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)
22 [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);
23 [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)
24 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821);
25 <https://www.drupal.org/docs/8/core/modules/media/overview>;
26 https://www.drupal.org/project/media_entity.

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

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

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

12 50. The presence of the above referenced elements are demonstrated, by way
13 of example, by reference to publicly available information. *See, e.g.*,
14 <https://www.drupal.org/project/ckeditor>;
15 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
16 [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)
17 [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);
18 https://www.drupal.org/docs/8/core/modules/custom_block/overview;
19 <https://www.drupal.org/docs/8/core/modules/contact/overview>.

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

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

26 53. Claim 9 recites the apparatus of claim 2, wherein said elements include a
27 button or an images, wherein said selectable settings include the selection of an
28

1 element style, and wherein said build tool includes means for storing information
2 representative of selected style in a database.

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

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

20 56. The Accused Instrumentalities infringe claim 10 of the '397 patent
21 through a combination of features which collectively practice each limitation of claim
22 10. For example, buttons can have multiple object states. *See, e.g.*,
23 <https://www.drupal.org/docs/8/core/themes/seven-theme>;
24 <https://wordpress.org/plugins/animate-everything/>.

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

27 58. The Accused Instrumentalities infringe claim 11 of the '397 patent
28 through a combination of features which collectively practice each limitation of claim

1 11. By way of example, the Accused Instrumentalities support CSS architecture. *See,*
2 *e.g.*, <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
3 *see also, e.g.*, <http://demos.dojotoolkit.org/demos/css3/demo.html>;
4 <https://wordpress.org/plugins/animate-everything/>.

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

10 60. The Accused Instrumentalities infringe claim 14 of the '397 patent
11 through a combination of features which collectively practice each limitation of claim
12 14.

13 61. By way of example, the Accused Instrumentalities include various CSS
14 libraries that are used extensively for adding transformations and timelines to selected
15 elements. *See, e.g.*, [https://www.drupal.org/docs/develop/standards/css/css-](https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8)
16 [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>.

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

19 63. The Accused Instrumentalities infringe claim 15 of the '397 patent
20 through a combination of features which collectively practice each limitation of claim
21 15.

22 64. By way of example, the Accused Instrumentalities enable descriptions of
23 elements describing CSS animations. *See, e.g.*,
24 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
25 <http://demos.dojotoolkit.org/demos/css3/demo.html>.

26 65. Claim 17 recites the apparatus of claim 2, wherein one or more of said
27 elements is a button or an image, wherein said description of elements is a transition,
28

1 an animation or a timeline, and wherein said build engine includes means to
2 synchronize said description of said one or more elements.

3 66. The Accused Instrumentalities infringe claim 17 of the '397 patent
4 through a combination of features which collectively practice each limitation of claim
5 17.

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

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

14 69. The Accused Instrumentalities infringe claim 20 of the '397 patent
15 through a combination of features which collectively practice each limitation of claim
16 20.

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

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

24 72. The Accused Instrumentalities infringe claim 24 of the '397 patent
25 through a combination of features which collectively practice each limitation of claim
26 24.

27 73. By way of example, the Accused Instrumentalities include two
28 customized runtime files, an HTML file and a second unique CSS file. *See, e.g.,*

1 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
2 <https://www.drupal.org/docs/8/system-requirements/web-server>;
3 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
4 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
5 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
6 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>.

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

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

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

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

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

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

1 redefine a size of a web page upon being displayed. *See, e.g.*,
2 http://www.w3schools.com/html/html_responsive.asp;
3 <https://www.drupal.org/docs/8/mobile/responsive-web-design>;
4 <https://www.drupal.org/docs/8/mobile/web-based-mobile-apps>.

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

19 81. The Accused Instrumentalities infringe claim 37 of the '397 patent
20 through a combination of features which collectively practice each limitation of claim
21 37. By way of example, modern internet browsers such as Microsoft Internet
22 Explorer, Mozilla's Firefox, Apple Safari, Google Chrome, and Opera include virtual
23 machines within the meaning of the '397 patent. (*See, e.g.*,
24 <http://developer.telerik.com/featured/a-guide-to-javascript-engines-for-idiots/>;
25 <http://dictionary.reference.com/browse/virtual+machine?s=t>). The Accused
26 Instrumentalities support the use of the latest versions of Internet Explorer 11 or later,
27 Microsoft Edge, latest-1, Firefox latest, latest-1, Chrome latest, latest-1, Safari latest,
28 latest-1 (Mac OS), Safari Mobile for iPad 2, iPad Mini, iPad with Retina Display (iOS

1 7 or later), for desktop site, Safari Mobile for iPhone 4 or later; iOS 7 or later, for
2 mobile site, Chrome for mobile latest-1 (Android 4 or later) for mobile site, where
3 *latest-1* means one major version earlier than the latest released version. (See
4 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
5 [http://devdocs.magento.com/guides/v2.0/install-gde/system-](http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html)
6 [requirements_browsers.html](http://themeforest.net/category/wordpress); <http://themeforest.net/category/wordpress>.)

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

1 area, a URL assigned by clicking and dragging over the text object you wish to link,
2 and then selecting the “chain” link button from the control bar; which opens a tabbed
3 panel where the user can designate the URL, target, etc. The text and its style are
4 saved to the database as part of the HTML of the description record.

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

7 84. The presence of the above referenced elements are demonstrated, by way
8 of example, by reference to publicly available information. *See, e.g.*,
9 <https://www.drupal.org/home>; <https://www.drupal.org/docs/8/system-requirements/browser-requirements>; <https://www.drupal.org/project/ckeditor>;
10 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
11 <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);
12 https://www.drupal.org/project/save_draft;
13 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
14 <https://www.drupal.org/docs/8/system-requirements/web-server>;
15 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
16 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
17 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
18 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
19 [http://docs.magento.com/m1/ee/user_guide/system-operations/index-](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html)
20 [management.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html); [http://docs.magento.com/m1/ee/user_guide/design/layout-](http://docs.magento.com/m1/ee/user_guide/design/layout-updates.html)
21 [updates.html](http://docs.magento.com/m1/ee/user_guide/system-operations/cache-page.html?Highlight=database%20retrieval); [cache-](http://docs.magento.com/m1/ee/user_guide/system-operations/cache-page.html?Highlight=database%20retrieval)
22 [page.html?Highlight=database%20retrieval](http://docs.magento.com/m1/ee/user_guide/system-operations/cache-page.html?Highlight=database%20retrieval);
23 http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage.html;
24 [http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage-](http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage-database.html)
25 [database.html](http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage-database.html);
26 http://docs.magento.com/m1/ee/user_guide/Resourcs/pdf/magento_enterprise_edition
27
28

1 _user_guide.pdf; http://docs.magento.com/m1/ee/user_guide/store-operations/stores-
2 [multiple.html](http://docs.magento.com/m1/ee/user_guide/store-operations/store-); http://docs.magento.com/m1/ee/user_guide/store-operations/store-
3 [hierarchy.html](http://docs.magento.com/m1/ee/user_guide/system-); http://docs.magento.com/m1/ee/user_guide/system-
4 [operations/browser-capabilities-detection.html](http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html);
5 http://docs.magento.com/m1/ce/user_guide/design/page-layout.html;
6 http://docs.magento.com/m1/ce/user_guide/design/layout-updates.html; and
7 http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition
8 [_user_guide.pdf](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition). Regarding Wordpress, *see, e.g.*,
9 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
10 http://codex.wordpress.org/Template_Hierarchy;
11 http://codex.wordpress.org/Function_Reference/the_title;
12 http://codex.wordpress.org/Function_Reference/the_content;
13 http://codex.wordpress.org/Template_Tags/get_the_title;
14 http://codex.wordpress.org/Query_Overview;
15 <https://www.wpbeginner.com/glossary/database/>; and
16 <https://codex.wordpress.org/Pages>.

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

20 86. Defendant was made aware of the '397 patent and its infringement
21 thereof at least as early as the filing of this Complaint.

22 87. Since the date of the filing of this Complaint, Defendant's infringement
23 of the '397 patent has been willful.

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

27 89. Plaintiff has been harmed by Defendant's infringing activities.
28

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

2 90. The allegations set forth in the foregoing paragraphs 1 through 89 are
3 incorporated into this Second Claim for Relief.

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

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

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

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

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

26 96. Accordingly, each claim of the '168 patent recites a combination of
27 elements sufficient to ensure that the claim in practice amounts to significantly more
28 than a patent on an ineligible concept.

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

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

8 99. Upon information and belief, Defendant has and continues to directly
9 infringe at least claims 1-6 of the '168 patent by using a browser-based website and/or
10 web page authoring tool in which the user-selected settings representing website
11 elements are stored in a database, and retrieval of said information to generate said
12 website (the "Accused Instrumentalities"). The Accused Instrumentalities include but
13 are not limited website building tools used and/or provided by Defendant, such as, for
14 example Drupal, Magento and/or Wordpress. *See, e.g.*,
15 <https://www.ampagency.com/livenation-insomniac-entertainment-media-case-study>;
16 <https://www.ampagency.com/sena-cases-accessories-case-study>; and
17 <https://www.ampagency.com/mobile-web-development-and-web-design-agency>

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

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

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

7 102. Further, by way of example, the JSON strings that are used by the
8 Accused Instrumentalities to generate, in part, element formatting originate from the
9 database and therefore reflect the database structure and contents showing, on
10 information and belief, the implementation of a multidimensional array structured
11 database comprising the objects that comprise the web site. By way of further
12 evidence, the JSON strings show that there are dimensions for the pages, for arrays of
13 columns, for arrays of sections, and for arrays of modules generated using the
14 Accused Instrumentalities. *See, e.g.*,
15 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>
16

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

23 104. By way of example, for the completed web site, the Accused
24 Instrumentalities include runtime files, such as, for example HTML CSS files. *See*,
25 *e.g.*, <https://www.drupal.org/home>; <https://www.drupal.org/docs/8/system-requirements/browser-requirements>; <https://www.drupal.org/project/ckeditor>;
26 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
27 <https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive->
28

1 images/10/03/2016/9821; Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
2 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
3 <https://www.drupal.org/docs/8/system-requirements/web-server>;
4 <https://www.drupal.org/docs/8/core/modules/rest/overview>;
5 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
6 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
7 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>;
8 <https://www.drupal.org/docs/8/core/modules/media/overview>;
9 https://www.drupal.org/project/media_entity;
10 <https://www.drupal.org/docs/8/core/modules/image/working-with-images>;
11 <http://demos.dojotoolkit.org/demos/css3/demo.html>;
12 <https://www.drupal.org/files/issues/Field.png>;
13 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>; https://www.drupal.org/project/save_draft.

15 105. Claim 2 of the '168 patent generally recites the system of claim 1,
16 wherein one of said plurality of objects is a child, and wherein the build engine is
17 configured to accept user input to associate a style with child button and child image
18 objects.

19 106. The Accused Instrumentalities infringe claim 2 of the '168 patent
20 through a combination of features which collectively practice each limitation of claim
21 2.

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

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

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

4 110. By way of example, the Accused Instrumentalities incorporate various
5 CSS libraries, and CSS-animations and CSS-transitions are used extensively for
6 adding transformations and timelines to selected elements. On information and belief,
7 this includes timelines for child buttons and child image objects. *See*
8 [http://docs.magento.com/m1/ee/user_guide/cms/banner-](http://docs.magento.com/m1/ee/user_guide/cms/banner-rotator.html?Highlight=carousel)
9 [rotator.html?Highlight=carousel](http://docs.magento.com/m1/ee/user_guide/cms/banner-rotator.html?Highlight=carousel).

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

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

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

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

20 115. On information and belief, the Accused Instrumentalities infringe claim 5
21 of the '168 patent through a combination of features which collectively practice each
22 limitation of claim 5. *See* [http://docs.magento.com/m1/ee/user_guide/design/merge-](http://docs.magento.com/m1/ee/user_guide/design/merge-css.html?Highlight=css)
23 [css.html?Highlight=css](http://docs.magento.com/m1/ee/user_guide/design/merge-css.html?Highlight=css).

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

27 117. The Accused Instrumentalities infringe claim 6 of the '168 patent through
28 a combination of features which collectively practice each limitation of claim 6. A

1 review of the API documentation behind websites created using the Accused
2 Instrumentalities reveals data that is stored as one or more of a Boolean, an integer, or
3 a string. *See, e.g.*, [https://www.drupal.org/docs/8/api/entity-api/defining-and-using-](https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions)
4 [content-entity-field-definitions](https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions).

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

8 119. Defendant was made aware of the '168 patent and its infringement
9 thereof at least as early as the filing of this Complaint.

10 120. Since the date of the filing of this Complaint, Defendant's infringement
11 of the '168 patent has been willful.

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

15 122. Plaintiff has been harmed by Defendant's infringing activities.

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

17 123. The allegations set forth in the foregoing paragraphs 1 through 122 are
18 incorporated into this Third Claim for Relief.

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

23 125. The inventions of the '287 patent resolve technical problems related to
24 generating content on a display of a device, such as the display of a mobile device.
25 For example, the inventions of the '287 patent feature a registry and an authoring tool
26 or Player configured to define a User Interface ("UI") object for display on the device,
27 where the UI object corresponds to a web component. Each UI object is either: 1)
28 selected by a user or 2) automatically selected by the system as a preferred UI object

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

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

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

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

27
28

1 129. Accordingly, each claim of the '287 patent recites a combination of
2 elements sufficient to ensure that the claim in practice amounts to significantly more
3 than a patent on an ineligible concept.

4 130. Plaintiff is the assignee and owner of the right, title and interest in and to
5 the '287 patent, including the right to assert all causes of action arising under the
6 patents and the right to any remedies for infringement of them.

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

26 132. In particular, claim 1 of the '287 patent recites 1 a system for generating
27 code to provide content on a display of a device, the system comprising: computer
28 memory storing a registry of: a) symbolic names required for evoking one or more

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

1 133. The Accused Instrumentalities infringe claim 1 of the '287 patent through
 2 a combination of features The Accused Instrumentalities infringe claim 1 of the '287
 3 patent through a combination of features which collectively practice each limitation of
 4 claim 1. By way of example, the Accused Instrumentalities feature a system for
 5 generating code to provide content on a display of a device. The system includes a
 6 server hosting the WordPress platform, which provides WordPress's WYSIWYG
 7 visual effects editor, and which is accessed through a WordPress-compatible browser.
 8 WordPress's WYSIWYG visual effects editor generates code, such as JavaScript or
 9 HTML code, for such options as defining title, text, images, videos and paragraph
 10 styles, while the browser displays the resulting content as a WordPress webpage on a
 11 display of a device.

12 134. For example, on information and belief, WordPress uses a variety of
 13 databases in its technology stack including MySQL. Data from the wp_options table
 14 for the website header and the data from the wp_posts table for the "WordPress Info"
 15 web page extracted directly from the Bitnami WordPress server-side database using
 16 MySQL Workbench. The stored data in the wp_options table includes the website's
 17 url, the website's title (blogname), the website's tagline (blogdescription), and the
 18 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

19
 20
 21
 22
 23
 24 The stored data in the wp_posts table for the "WordPress Info" web page includes
 25 information corresponding to user selected settings such as, for example, the color red
 26 ("#ff0000") for "manages 22%". Other user selections shown in stored database data
 27 below include, for ex-ample, the image filename for the image
 28 (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the image

1 alignment (class="wp-image-24 aligncenter"), and a selected paragraph style (h3) for
2 "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;">http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png</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

14
15 WordPress's HTML, CSS, Java, and JSON coding capabilities further are shown, e.g.,
16 by <http://codex.wordpress.org/Templates>, <https://codex.wordpress.org/CSS>,
17 <https://developer.wordpress.org/rest-api/>.

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

22
23 136. The computer memory stores a registry of a) symbolic names required
24 for evoking one or more web components each related to a set of inputs and outputs of
25 a web service obtainable over a network, where the symbolic names are character
26 strings that do not contain either a persistent address or pointer to an output value
27 accessible to the web service. The WordPress MySQL database contains symbolic
28 names required for evoking one or more web components each related to a set of

1 inputs and outputs of a web service obtainable over a network by the formatting of the
2 symbolic names in conjunction with WordPress's WYSIWYG visual effects editor,
3 widget, and plugin authoring tools.

4 137. Furthermore, each symbolic name has an associated data format class
5 type corresponding to a subclass of defined UI objects *i.e.*, element/UI components,
6 that supports the data format type of the symbolic name, and has a preferred UI object
7 as evidenced by the JSON formatting of the name in conjunction with WordPress'
8 WYSIWYG visual effects editor and widget authoring tools. JSON names are strings
9 that only represent the symbolic names that are bound both to a web service input
10 and/or output and to a UI object. All JSON names in the name/value pairs are
11 character strings. WordPress' WYSIWYG visual effects editor includes elements for
12 defining the layout for placement of the defined UI objects. Widgets, plug-ins and
13 other elements correspond to the defined UI objects and are the product of the JSON
14 formatting. (*See*, for example,
15 https://codex.wordpress.org/WordPress_Lessons#Template_Files,
16 https://codex.wordpress.org/WordPress_Widgets,
17 https://codex.wordpress.org/Plugin_Resources, and
18 https://codex.wordpress.org/Plugin_API.)

19 138. The computer memory also stores b) an address of the web service.
20 Because WordPress contains web services, it contains the corresponding addresses for
21 the web services. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
22 https://codex.wordpress.org/WordPress_Widgets,
23 https://codex.wordpress.org/Plugin_Resources, and
24 https://codex.wordpress.org/Plugin_API.)

25 139. The Accused Instrumentalities feature an authoring tool in the form of
26 WordPress's WYSIWYG visual effects editor, widget, and plug-in authoring tools.
27 (*See*, for example, https://codex.wordpress.org/WordPress_Widgets,

28

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

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

11 141. Each defined UI object is either: 1) selected by a user of the authoring
12 tool; or 2) automatically selected by the system as the preferred UI object
13 corresponding to the symbolic name of the web component selected by the user of the
14 authoring tool. WordPress's UI objects are automatically selected by the system as
15 the preferred UI object corresponding to the symbolic name of the web component
16 selected by the user of the authoring tool, i.e., a UI object selected by a user is
17 automatically selected. When a Widget is selected in the WordPress Widget selection
18 list steps 1 to 5 under "Displaying Widgets", the widget UI will automatically
19 displayed in the Web Page Sidebar. (See, for example,

20 https://codex.wordpress.org/WordPress_Widgets,
21 https://codex.wordpress.org/Plugin_Resources,
22 https://codex.wordpress.org/Plugin_API.)

23 142. The authoring tool is configured to access the computer memory to select
24 the symbolic name corresponding to the web component of the defined UI object by a
25 JSON formatted element.

26 143. The authoring tool is also configured to associate the selected symbolic
27 name with the defined UI object, i.e., the JSON formatted element, where the selected
28 symbolic name is only available to UI objects that support the defined data format

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

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

19 145. The WordPress authoring tool is further configured to produce a Player
20 (*see*, for example, <https://codex.wordpress.org/CSS2>,
21 https://codex.wordpress.org/Using_Javascript, [https://developer.wordpress.org/rest-](https://developer.wordpress.org/rest-api/)
22 [api/](https://developer.wordpress.org/rest-api/)), where the Player is a device-dependent code. WordPress contains a Player in
23 the form of a runtime player. The Accused Instrumentality produces a device
24 dependent file, which is wrapped inside the runtime file. In order for a site to display
25 on different devices through a browser or through responsive capabilities, there is
26 device dependent code (*see*, for example, [https://torquemag.io/2017/08/make-](https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/)
27 [wordpress-website-mobile-friendly/](https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/) and <https://wordpress.org/themes/ultra/>.)

28

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

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

26 148. The Accused Instrumentalities feature a system where the web service
27 utilizes the input symbolic name and the user provided one or more input values for
28 generating one or more output values having an associated output symbolic name.

1 Because of the JSON formatting, the output values having an associated output
2 symbolic name. (See, for example, <https://developer.wordpress.org/rest-api/>,
3 <https://developer.wordpress.org/rest-api/reference/>,
4 https://codex.wordpress.org/Widgets_API,
5 https://codex.wordpress.org/Plugin_Resources, and
6 https://codex.wordpress.org/Plugin_API.)

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

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

27
28

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

4 152. The Accused Instrumentalities infringe claim 2 of the '287 patent through
5 a combination of features which collectively practice each limitation of claim 1. By
6 way of example, the registry includes definitions of input and output related to a web
7 service as evidenced by WordPress's JSON formatting characteristics of the defined
8 UI objects. (See, for example, <https://developer.wordpress.org/rest-api/>;
9 <https://developer.wordpress.org/rest-api/reference/>;
10 <https://developer.wordpress.org/rest-api/reference/posts/>
11 https://developer.wordpress.org/rest-api/reference/posts/#schema-title_,
12 https://codex.wordpress.org/Widgets_API;
13 https://codex.wordpress.org/Plugin_Resources; and
14 https://codex.wordpress.org/Plugin_API;
15 https://codex.wordpress.org/WordPress_Widgets_.)

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

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

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

27 156. The Accused Instrumentalities infringe claim 4 of the '287 patent through
28 a combination of features which collectively practice each limitation of claim 4. By

1 way of example, the Accused Instrumentalities additionally feature a defined UI
2 object that is an input field for a chat.

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

6 158. The Accused Instrumentalities infringe claim 5 of the '287 patent through
7 a combination of features which collectively practice each limitation of claim 1. By
8 way of example, the Accused Instrumentalities additionally feature a defined UI
9 object that is an input field for a web service.

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

13 160. The Accused Instrumentalities infringe claim 11 of the '287 patent
14 through a combination of features which collectively practice each limitation of claim
15 11. WordPress sends all files over a network using a variety of databases in its
16 technology stack including MySQL. These backend capabilities provided the code
17 over a network. By way of example, data from the wp_options table for the website
18 header and from the wp_posts table for the "WordPress Info" web page are extracted
19 directly from the Bitnami WordPress server-side database using MySQL Workbench.
20 The stored data in the wp_options table includes the website's url, the website's title
21 (blogname), the website's tagline (blogdescription), and the active template (style
22 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
 2 information corresponding to user selected settings such as, for example, the color red
 3 (“#ff0000”) for “manages 22%”. Other user selections shown in stored database data
 4 below include, for ex-ample, the image filename for the image
 5 (http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png), the image
 6 alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for
 7 “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

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

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

26 162. The Accused Instrumentalities infringe claim 12 of the ’287 patent
 27 through a combination of features which collectively practice each limitation of claim
 28

1 1. By way of example, the Accused Instrumentalities' incorporation of WordPress
2 includes widgets. (See, for example., https://codex.wordpress.org/Widgets_API.)

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

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

23 165. The Accused Instrumentalities feature a Player, where the Player is a
24 device-dependent code. The device has a Player (*see*, for example,
25 <https://codex.wordpress.org/CSS>, https://codex.wordpress.org/Using_Javascript,
26 <https://developer.wordpress.org/rest-api/>) in the form of a runtime player that is a
27 device dependent code. WordPress produces a device dependent file, which is
28 wrapped inside a runtime file. In order for a site to display on different devices

1 through a browser or through responsive capabilities, there is device dependent code.
2 (See, for example, [https://torquemag.io/2017/08/make-wordpress-website-mobile-](https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/)
3 [friendly/](https://wordpress.org/themes/ultra/); <https://wordpress.org/themes/ultra/>.)

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

24 167. Each web component includes a plurality of symbolic names of inputs
25 and outputs associated with each web service. The plurality of symbolic names of
26 inputs and outputs associated with each web service is a feature of their JSON
27 formatting characteristics. Each symbolic name has an associated data format class
28 type corresponding to a subclass of UI objects that supports the data format type of the

1 symbolic name, and has a preferred UI object as demonstrated by the presence of
2 JSON formatting in conjunction with WordPress' WYSIWYG and widget
3 capabilities. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
4 <https://developer.wordpress.org/rest-api/>,
5 https://codex.wordpress.org/WordPress_Widgets,
6 https://codex.wordpress.org/Plugin_API, and
7 https://codex.wordpress.org/Plugin_Resources.)

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

21 <https://developer.wordpress.org/rest-api/>,
22 https://codex.wordpress.org/WordPress_Widgets,
23 https://codex.wordpress.org/Plugin_Resources,
24 https://codex.wordpress.org/Plugin_API.)

25 169. The registry also includes b) an address of the web service. Because
26 WordPress contains web services, it contains the corresponding web addresses.
27 Because WordPress contains web services, both as a library of Widgets (*see*
28 https://codex.wordpress.org/Widgets_API) and Plug-ins (*see*

1 https://codex.wordpress.org/Plugin_API), it contains the corresponding addresses of
2 the web services. All web services are represented as a wsdl (Web Service
3 Description Language) and wsdl's have URLs to point to the internet location that
4 receives the web service's inputs and returns the web service's outputs (*see*
5 <https://www.soapui.org/soap-and-wsdl/working-with-wsdl.html>).

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

18 171. The Accused Instrumentalities include selecting the symbolic name from
19 the web component (i.e. WordPress Widget or Plug-in) corresponding to the defined
20 UI object, where the selected symbolic name has an associated data format class type
21 corresponding to a subclass of UI objects that support the data format type of the
22 symbolic name and has the preferred UI object. WordPress accesses its memory to
23 select the symbolic name corresponding to the web component of the defined UI
24 object (as evidenced by JSON data formatting), associate the selected symbolic name
25 with the defined UI object (the JSON element corresponding to an element), where the
26 selected symbolic name is only available to UI objects that support the defined data
27 format associated with that symbolic name (the element associated with at JSON
28 string). (*See*, for example, <https://developer.wordpress.org/rest-api/>,

1 <https://developer.wordpress.org/rest-api/reference/>,
2 https://codex.wordpress.org/WordPress_Widgets,
3 https://codex.wordpress.org/Plugin_API, and
4 https://codex.wordpress.org/Plugin_Resources.) Additionally, the preferred UI object
5 is the selected UI object. JSON names are strings that only represent the symbolic
6 names that are bound both to a web service input and/or output and to a UI object. All
7 JSON names in the name/value pairs are character strings. When the WordPress
8 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. (*See*, for example,
10 <https://developer.wordpress.org/rest-api/>,
11 https://codex.wordpress.org/WordPress_Widgets,
12 https://codex.wordpress.org/Plugin_Resources,
13 https://codex.wordpress.org/Plugin_API.)

14 172. The Accused Instrumentalities include associating the selected symbolic
15 name with the defined UI object.

16 173. The Accused Instrumentalities include producing an Application
17 including the selected symbolic name of the defined UI object, where the Application
18 is a device-independent code. WordPress produces an Application included in the
19 symbolic name of the defined UI object. (*See*, for example,
20 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
21 https://codex.wordpress.org/WordPress_Widgets,
22 https://codex.wordpress.org/Plugin_API, and
23 https://codex.wordpress.org/Plugin_Resources.)

24 174. The Accused Instrumentalities feature a system where the Application
25 and Player are provided to the device and executed on the device and when the user of
26 the device provides one or more input values associated with an input symbolic name
27 to an input of the defined UI object. Because the Accused Instrumentalities
28 incorporate a system that includes WordPress, when a user of the device provides one

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

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

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

28

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

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

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

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

26 180. The Accused Instrumentalities infringe claim 16 of the '287 patent
27 through a combination of features which collectively practice each limitation of claim
28 16. By way of example, the Accused Instrumentalities include definitions of input

1 and output related to a web service as based on their JSON formatting characteristics
2 and the defined UI object functionality. *See, e.g.,* [https://developer.wordpress.org/rest-](https://developer.wordpress.org/rest-api/)
3 [api/](https://developer.wordpress.org/rest-api/reference/); <https://developer.wordpress.org/rest-api/reference/>;
4 <https://developer.wordpress.org/rest-api/reference/posts/>
5 <https://developer.wordpress.org/rest-api/reference/posts/#schema-title>
6 https://codex.wordpress.org/Widgets_API;
7 https://codex.wordpress.org/Plugin_Resources; and
8 https://codex.wordpress.org/Plugin_API;
9 https://codex.wordpress.org/WordPress_Widgets.

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

13 182. The Accused Instrumentalities infringe claim 17 of the '287 patent
14 through a combination of features which collectively practice each limitation of claim
15 17. By way of example, the Accused Instrumentalities feature web components
16 including web chat, Reuters RSS feed, Calendar image, and map image widgets.

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

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

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

27 186. The Accused Instrumentalities infringe claim 19 of the '287 patent
28 through a combination of features which collectively practice each limitation of claim

19. By way of example, the Accused Instrumentalities additionally feature a UI object that is an input field for a web service.

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

188. The Accused Instrumentalities infringe claim 25 of the '287 patent through a combination of features which collectively practice each limitation of claim 25. WordPress sends all files over a network using a variety of databases in its technology stack including MySQL. These backend capabilities provided the code over a network. By way of example, data from the wp_options table for the website header and from the wp_posts table for the “WordPress Info” web page are 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 example, 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;"><a ><="" ><img="" a><="" alt="icon2" class="wp-image-24 aligncenter" height="36" href="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" p="" p><="" src="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" width="136"> <p>&nbsp;</p> <p>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</p> <p>&nbsp;</p> </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.

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

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

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

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

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

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

6 195. Plaintiff has been harmed by Defendant's infringing activities.

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

8 196. The allegations set forth in the foregoing paragraphs 1 through 195 are
9 incorporated into this Fourth Claim for Relief.

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

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

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

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

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

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

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

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

19 204. Upon information and belief, Defendant has and continues to directly
20 infringe at least claims 1-5, 11, 12, 15-19, 25 and 26 of the '044 patent by a system
21 which includes a computer memory and an authoring tool or Player configured to
22 define a User Interface ("UI") object for display on the device, where the UI object
23 corresponds to a web component and where each UI object is either: 1) selected by a
24 user or 2) automatically selected by the system as a preferred UI object corresponding
25 to a symbolic name of the web component. Additionally, the computer memory and
26 the authoring tool or Player are configured to build an Application consisting of one or
27 more web page views to provide for the display of at least a portion of one or more of
28 the web pages (the "Accused Instrumentalities"). The Accused Instrumentalities

1 include platforms that enable the functionality described above and include but are not
2 limited to, for example, WordPress. *See, e.g.*, [https://www.ampagency.com/mobile-](https://www.ampagency.com/mobile-web-development-and-web-design-agency)
3 [web-development-and-web-design-agency](https://www.ampagency.com/mobile-web-development-and-web-design-agency).

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

1 the one or more web pages, wherein when the Application and Player are provided to
2 the device and executed on the device, and when the user of the device provides one
3 or more input values associated with an input symbolic name to an input of the
4 defined UI object, the device provides the user provided one or more input values and
5 corresponding input symbolic name to the web service, the web service utilizes the
6 input symbolic name and the user provided one or more input values for generating
7 one or more output values having an associated output symbolic name, and the Player
8 receives the output symbolic name and corresponding one or more output values and
9 provides instructions for the display of the device to present an output value in the
10 defined UI object.

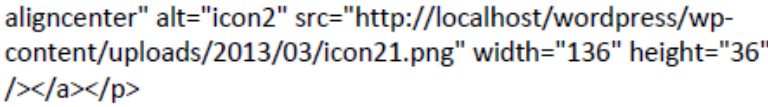
11 206. The Accused Instrumentalities infringe claim 1 of the '044 patent through
12 a combination of features which collectively practice each limitation of claim 1. By
13 way of example, the Accused Instrumentalities feature a system for generating code to
14 provide content on a display of a device. The system includes a WordPress server,
15 which provides WordPress's WYSIWYG visual effects editor and a WordPress-
16 compatible browser. WordPress's WYSIWYG visual effects editor generates code,
17 such as JavaScript or HTML code for such as options for defining title, text, images,
18 videos and paragraph styles, while the browser displays the resulting content as a
19 WordPress webpage on a display of a device, such as a computer display.

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

27
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 style="text-align: center;">width="136" height="36"/></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, e.g., by <http://codex.wordpress.org/Templates>, <https://codex.wordpress.org/CSS>, <https://developer.wordpress.org/rest-api/>.

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

6 209. The computer memory stores a) symbolic names required for evoking
7 one or more web components each related to a set of inputs and outputs of a web
8 service obtainable over a network, where the symbolic names are character strings that
9 do not contain either a persistent address or pointer to an output value accessible to the
10 web service. The WordPress MySQL database contains symbolic names required for
11 evoking one or more web components each related to a set of inputs and outputs of a
12 web service obtainable over a network as demonstrated by the formatting in
13 conjunction with WordPress's WYSIWYG visual effects editor, widget, and plugin
14 authoring tools. JSON names are strings that only represent the symbolic names that
15 are bound both to a web service input and/or output and to a UI object. All JSON
16 names in the name/value pairs are character strings. WordPress' WYSIWYG visual
17 effects editor includes elements for defining the layout for placement of the defined
18 UI objects. Widgets, plug-ins and other elements correspond to the defined UI objects
19 and are the product of the JSON formatting. (*See*, for example,
20 https://codex.wordpress.org/WordPress_Lessons#Template_Files,
21 https://codex.wordpress.org/WordPress_Widgets,
22 https://codex.wordpress.org/Plugin_Resources, and
23 https://codex.wordpress.org/Plugin_API.)

24 210. Furthermore, each symbolic name has an associated data format class
25 type corresponding to a subclass of UI objects that supports the data format type of the
26 symbolic name, and has a preferred UI object as demonstrated by the presence of
27 JSON formatting characteristics in conjunction with WordPress' WYSIWYG visual
28 effects editor and widget authoring tools. JSON names are strings that only represent

1 the symbolic names that are bound both to a web service input and/or output and to a
2 UI object. All JSON names in the name/value pairs are character strings. WordPress'
3 WYSIWYG visual effects editor includes elements for defining the layout for
4 placement of the defined UI objects. Widgets, plug-ins and other elements correspond
5 to the defined UI objects and are the product of the JSON formatting. (*See*, for
6 example, https://codex.wordpress.org/WordPress_Lessons#Template_Files,
7 https://codex.wordpress.org/WordPress_Widgets,
8 https://codex.wordpress.org/Plugin_Resources, and
9 https://codex.wordpress.org/Plugin_API.)

10 211. The computer memory also stores b) an address of the web service.
11 Because WordPress contains web services, it contains the corresponding addresses for
12 the web services. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
13 https://codex.wordpress.org/WordPress_Widgets,
14 https://codex.wordpress.org/Plugin_Resources, and
15 https://codex.wordpress.org/Plugin_API.)

16 212. The Accused Instrumentalities feature an authoring tool in the form of
17 WordPress's WYSIWYG visual effects editor, widget, and plug-in authoring tools.
18 (*See*, for example, https://codex.wordpress.org/WordPress_Widgets,
19 https://codex.wordpress.org/Plugin_Resources,
20 https://codex.wordpress.org/Plugin_API.)

21 213. The authoring tool is configured to define a UI object for presentation on
22 the display, where the defined UI object corresponds to a web component included in
23 the computer memory selected from a group consisting of an input of the web service
24 and an output of the web service. WordPress's WYSIWYG visual effects editor and
25 widget authoring tools define the presence of a UI object for presentation on a display
26 and the defined UI object corresponds to a web component included in the computer
27 memory selected from a group consisting of an input of the web service and an output
28 of the web service.

1 214. Each defined UI object is either: 1) selected by a user of the authoring
2 tool; or 2) automatically selected by the system as the preferred UI object
3 corresponding to the symbolic name of the web component selected by the user of the
4 authoring tool. WordPress's UI objects are automatically selected by the system as
5 the preferred UI object corresponding to the symbolic name of the web component
6 selected by the user of the authoring tool, *i.e.*, a UI object selected by a user is
7 automatically selected. When a Widget is selected in the WordPress Widget selection
8 list (*See* https://codex.wordpress.org/WordPress_Widgets) steps 1 to 5 under
9 "Displaying Widgets", the widget UI will automatically displayed in the Web Page
10 Sidebar. (*See*, for example, https://codex.wordpress.org/WordPress_Widgets,
11 https://codex.wordpress.org/Plugin_Resources,
12 https://codex.wordpress.org/Plugin_API.)

13 215. The authoring tool is configured to access the computer memory to select
14 the symbolic name corresponding to the web component of the defined UI object
15 based on its JSON formatting characteristics.

16 216. The authoring tool is also configured to associate the selected symbolic
17 name with the defined UI object, *i.e.*, the JSON formatted element, where the selected
18 symbolic name is only available to UI objects that support the defined data format
19 associated with the element associated with that symbolic name, *i.e.*, JSON string.
20 JSON names are strings that only represent the symbolic names that are bound both to
21 a web service input and/or output and to a UI object. All JSON names in the
22 name/value pairs are character strings. When the WordPress Editor makes a UI
23 element request a JSON request is sent to the Server and a JSON data response is
24 provided to the Interface. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
25 https://codex.wordpress.org/WordPress_Widgets,
26 https://codex.wordpress.org/Plugin_Resources,
27 https://codex.wordpress.org/Plugin_API.)

28

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

6 218. The authoring tool is also configured to retrieve the information
7 representative of the one or more the UI object settings stored in the database based on
8 the JSON strings. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
9 <https://developer.wordpress.org/rest-api/reference/>,
10 https://codex.wordpress.org/Plugin_Resources,
11 https://codex.wordpress.org/Plugin_API, and
12 https://codex.wordpress.org/Widgets_API.)

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

27 220. The Accused Instrumentalities feature a system where the Application
28 and Player are provided to the device and executed on the device.

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

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

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

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

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

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

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

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

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

7 228. The Accused Instrumentalities infringe claim 3 of the '044 patent through
8 a combination of features which collectively practice each limitation of claim 3. By
9 way of example, the Accused Instrumentalities feature web components additionally
10 including web chat, Reuters RSS feed, Calendar image, and map image widgets.

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

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

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

22 232. The Accused Instrumentalities infringe claim 5 of the '044 patent through
23 a combination of features which collectively practice each limitation of claim 5. By
24 way of example, the Accused Instrumentalities additionally feature a defined UI
25 object that is an input field for a web service.

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

1 234. The Accused Instrumentalities infringe claim 11 of the '044 patent
 2 through a combination of features which collectively practice each limitation of claim
 3 11. WordPress sends all files over a network using a variety of databases in its
 4 technology stack including MySQL. These backend capabilities provided the code
 5 over a network. By way of example, data from the wp_options table for the website
 6 header and from the wp_posts table for the “WordPress Info” web page are extracted
 7 directly from the Bitnami WordPress server-side database using MySQL Workbench.
 8 The stored data in the wp_options table includes the website’s url, the website’s title
 9 (blogname), the website’s tagline (blogdescription), and the active template (style
 10 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

11
 12
 13
 14
 15 The stored data in the wp_posts table for the “WordPress Info” web page includes
 16 information corresponding to user selected settings such as, for example, the color red
 17 (“#ff0000”) for “manages 22%”. Other user selections shown in stored database data
 18 below include, for ex-ample, the image filename for the image
 19 (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the image
 20 alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for
 21 “Heading 3”.
 22
 23
 24
 25
 26
 27
 28

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).

235. 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.

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

237. The Accused Instrumentalities infringe claim 15 of the '044 patent through a combination of features which collectively practice each limitation of claim 15. By way of example, the Accused Instrumentalities feature a method of displaying content on a display of a device having a Player, in the form of a runtime player and a non-volatile computer memory storing the WordPress MySQL database functionality

1 on the device. The non-volatile computer memory stores symbolic names required for
2 evoking one or more web components each related to a set of inputs and outputs of a
3 web service obtainable over a network, where the symbolic names are character
4 strings that do not contain either a persistent address or pointer to an output value
5 accessible to the web service. The WordPress MySQL database contains symbolic
6 names required for evoking one or more web components each related to a set of
7 inputs and outputs of a web service obtainable over a network as demonstrated by the
8 formatting in conjunction with WordPress's WYSIWYG visual effects editor, widget,
9 and plugin authoring tools.

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

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

23 240. The Accused Instrumentalities include defining a UI object for
24 presentation on the display, where the UI object corresponds to a web component
25 included in the computer memory, where the web component is selected from a group
26 consisting of an input of a web service and an output of the web service. WordPress
27 defines a user interface object, *i.e.*, an element/UI component, for presentation on
28 display, where the UI object corresponds to a web component included in the non-

1 volatile computer memory selected from a group consisting of an input of a web
2 service and an output of the web service (as evidenced by JSON data formatting)..

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

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

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

25 244. The Accused Instrumentalities include building an Application consisting
26 of one or more web page views from at least a portion of the database utilizing the
27 Player, where the Player utilizes information stored in the database to generate for the
28

1 display of at least a portion of the one or more web pages. WordPress builds an
2 Application included in the symbolic name of the defined UI object.

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

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

1 <http://codex.wordpress.org/Templates>;

2 http://codex.wordpress.org/Template_Tags/get_the_title; and

3 http://codex.wordpress.org/Query_Overview.

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

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

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

25 250. The Accused Instrumentalities infringe claim 17 of the '044 patent
26 through a combination of features which collectively practice each limitation of claim
27 17. By way of example, the Accused Instrumentalities feature web components
28

1 additionally including web chat, Reuters RSS feed, Calendar image, and map image
2 widgets.

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

6 252. The Accused Instrumentalities infringe claim 18 of the '044 patent
7 through a combination of features which collectively practice each limitation of claim
8 18. By way of example, the Accused Instrumentalities additionally feature an UI
9 object that is an input field for a chat.

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

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

17 255. Claim 25 of the '044 patent recites a method of displaying content on a
18 display that includes all the elements of claim 15, additionally where the method
19 includes providing the code over the network.

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

25 257. Claim 26 of the '044 patent recites a method of displaying content on a
26 display that includes all the elements of claim 15, additionally where the UI object
27 corresponds to a widget.

28

1 The Accused Instrumentalities infringe claim 26 of the '044 patent through a
2 combination of features which collectively practice each limitation of claim 26. By
3 way of example, the Accused Instrumentalities employ WordPress which includes
4 widgets. See, e.g., https://codex.wordpress.org/Widgets_API.

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

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

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

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

15 262. Plaintiff has been harmed by Defendant's infringing activities.

16 **JURY DEMAND**

17 Pursuant to Rule 38 of the Federal Rules of Civil Procedure, Plaintiff demands
18 a trial by jury on all issues triable as such.

19 **PRAYER FOR RELIEF**

20 WHEREFORE, Plaintiff demands judgment for itself and against Defendant as
21 follows:

22 A. An adjudication that Defendant has infringed the '397, '168, '287, and
23 '044 patents;

24 B. An award of damages to be paid by Defendant adequate to compensate
25 Plaintiff for Defendant's past infringement of the '397, '168, '287, and '044 patents,
26 and any continuing or future infringement through the date such judgment is entered,
27 including interest, costs, expenses and an accounting of all infringing acts including,
28 but not limited to, those acts not presented at trial;

1 C. A declaration that this case is exceptional under 35 U.S.C. § 285, and an
2 award of Plaintiff's reasonable attorneys' fees; and

3 D. An award to Plaintiff of such further relief at law or in equity as the
4 Court deems just and proper.

5 Dated: June 13, 2019

6 By: /s/ Jeffrey Francis Craft
7 Jeffrey Francis Craft (SBN 147186)
8 jcraft@devlinlawfirm.com
9 DEVLIN LAW FIRM LLC
10 1731 Fox Springs Circle,
11 Newbury Park, CA 91320

12 *Attorneys for Plaintiff Express Mobile, Inc.*