

1 Seth W. Wiener (SBN 203747)
seth@sethwienerlaw.com
2 LAW OFFICES OF SETH WIENER
609 Karina Court
3 San Ramon, CA 94582
Telephone: (925) 487-5607
4

5 Jeffrey Francis Craft (SBN 147186)
jcraft@devlinlawfirm.com
DEVLIN LAW FIRM LLC
6 1731 Fox Springs Circle,
Newbury Park, CA 91320
7

8 Timothy Devlin (*pro hac vice* to be filed)
tdevlin@devlinlawfirm.com
9 Robert Kiddie (*pro hac vice* to be filed)
rkiddie@devlinlawfirm.com
DEVLIN LAW FIRM LLC
10 1526 Gilpin Avenue
Wilmington, DE 19806
11 Telephone: (302) 449-9010
Facsimile: (302) 353-4251
12

13 Attorneys for Express Mobile, Inc.

14 UNITED STATES DISTRICT COURT
15 NORTHERN DISTRICT OF CALIFORNIA
16 _____ DIVISION
17

18 EXPRESS MOBILE, INC.,
19

20 Plaintiff,
21

22 vs.
23

24 E-ZEST SOLUTIONS, INC.
25

26 Defendant.
27
28

) Case No.: 3:19-cv-03351
)
)
)
)

) **COMPLAINT FOR PATENT**
) **INFRINGEMENT**
)

) DEMAND FOR JURY TRIAL
)
)
)
)

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

3 **NATURE OF THE ACTION**

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

6 **THE PARTIES**

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

9 3. Upon information and belief, e-Zest is a corporation organized and existing under the
10 laws of Delaware, with a place of business at 1900 Camden Ave., Suite 101. San Jose, CA 95124
11 and can be served through its registered agent, Incorp Services, Inc., 919 N. Market Street, Suite
12 950, Wilmington, DE 19801.

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

17 **JURISDICTION AND VENUE**

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

6 21. Upon information and belief, Defendant has and continues to directly infringe at least
7 claims 1-6, 8-11, 14-15, 17, 20, 24-25, 35, and 37 of the '397 patent by using a browser-based
8 website and/or web page authoring tool in which the user-selected settings representing website
9 elements are stored in a database, and in which said stored information is retrieved to generate said
10 website (the "Accused Instrumentalities"). The Accused Instrumentalities include but are not
11 limited to the website building tools used and/or provided by Defendant, such as, for example
12 Drupal, Joomla, Magento and/or Wordpress. *See, e.g.*, [https://www.e-zest.com/case-study-drupal-](https://www.e-zest.com/case-study-drupal-based-corporate-blogging-and-website)
13 [based-corporate-blogging-and-website](https://www.e-zest.com/joomla-development-services); <https://www.e-zest.com/joomla-development-services>;
14 [https://www.e-zest.com/casestudy-enabling-unified-experience-with-headless-magento-and-aem-for-](https://www.e-zest.com/casestudy-enabling-unified-experience-with-headless-magento-and-aem-for-a-fortune-500-healthcare-giant)
15 [a-fortune-500-healthcare-giant](https://www.e-zest.com/casestudy-enabling-unified-experience-with-headless-magento-and-aem-for-a-fortune-500-healthcare-giant); [https://www.e-zest.com/case-study-transition-from-a-static-legacy-](https://www.e-zest.com/case-study-transition-from-a-static-legacy-site-to-wordpress-cms)
16 [site-to-wordpress-cms](https://www.e-zest.com/case-study-transition-from-a-static-legacy-site-to-wordpress-cms).

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

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

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

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

23 25. By way of further example, the Accused Instrumentalities enable users to produce
24 websites through browsers on users' computers via interaction with an Internet server. For example,
25 in order to add a new page to a user's website, the user logs in and then a server of the Accused
26 Instrumentalities initiates presentation to the user through a browser of a website-builder tool. From
27 the interface—sometimes referred to as a dashboard—of the Accused Instrumentalities, the user can
28 navigate and add elements and element properties commensurate with a new page. A display is

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

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

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

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

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

1 http://codex.wordpress.org/Function_Reference/the_title;
2 http://codex.wordpress.org/Function_Reference/the_content;
3 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
4 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
5 http://codex.wordpress.org/Query_Overview. Regarding Joomla, *see, e.g.*,
6 https://docs.joomla.org/Joomla_Browser_Support;
7 <https://www.joomlart.com/documentation/other/joomla-3-and-joomla-2-5-system-requirement>;
8 <https://showcase.joomla.org/>; https://docs.joomla.org/Editor_form_field_type;
9 <https://developer.joomla.org/coding-standards/html.html>; <https://developer.joomla.org/coding-standards/css.html>; <https://developer.joomla.org/coding-standards/javascript.html>;
10 https://docs.joomla.org/Generating_JSON_output; <https://api.joomla.org/cms-3/classes/Joomla.CMS.Input.Json.html>;
11 https://docs.joomla.org/How_do_you_assign_a_module_to_specific_pages%3F;
12 https://docs.joomla.org/Where_are_the_web_pages%3F.

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

23 31. The Accused Instrumentalities infringe claim 2 of the '397 patent through a
24 combination of features which collectively practice each limitation of claim 2. By way of example,
25 modern internet browsers such as Microsoft Internet Explorer, Mozilla's Firefox, Apple Safari,
26 Google Chrome, and Opera include virtual machines within the meaning of the '397 patent. (*See,*
27 *e.g.*, <http://developer.telerik.com/featured/a-guide-to-javascript-engines-for-idiots/>;
28 <http://dictionary.reference.com/browse/virtual+machine?s=t>). The Accused Instrumentalities

1 support the use of the latest versions of Internet Explorer 11 or later, Microsoft Edge, latest-1,
2 Firefox latest, latest-1, Chrome latest, latest-1, Safari latest, latest-1 (Mac OS), Safari Mobile for
3 iPad 2, iPad Mini, iPad with Retina Display (iOS 7 or later), for desktop site, Safari Mobile for
4 iPhone 4 or later; iOS 7 or later, for mobile site, Chrome for mobile latest-1 (Android 4 or later) for
5 mobile site, where *latest-1* means one major version earlier than the latest released version. (*See*,
6 *e.g.*, <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
7 http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html;
8 <http://themeforest.net/category/wordpress>; https://docs.joomla.org/Joomla_Browser_Support
9 http://devdocs.magento.com/guides/v2.0/install-gde/system-requirements_browsers.html.) All of
10 these browsers rely on browser engines comprising virtual machines to interpret and execute
11 JavaScript and HTML to render web pages on a computer.

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

26 33. Data is stored in a database, including information corresponding to user selected
27 settings such as, for example, the selections of text color. Other user selections are also stored
28 including, for example, the layout, image filenames, thumbnails, and paragraph margin settings for

1 defining the alignment of an image location. The Accused Instrumentalities build one or more web
2 pages to generate a website from at least a portion of a database and at least one run time file, where
3 at least one run time file utilizes information stored in said database to generate virtual machine
4 commands for the display of at least a portion of said one or more web pages.

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

12 35. Additionally, the "PHP code," including the PHP template files, can be viewed in the
13 file directory for the Accused Instrumentalities, and this directory includes various other runtime
14 files (including other PHP files, JavaScript files, PHTML, and XML). *See, e.g.*,
15 <https://techterms.com/definition/runtime>.

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

19 37. The presence of the above referenced elements are demonstrated, by way of example,
20 by reference to publicly available information. Regarding Drupal, *see, e.g.*,
21 <https://www.drupal.org/home>; [https://www.drupal.org/docs/8/system-requirements/browser-](https://www.drupal.org/docs/8/system-requirements/browser-requirements)
22 [requirements; https://www.drupal.org/project/ckeditor](https://www.drupal.org/project/ckeditor);
23 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
24 [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)
25 [images/10/03/2016/9821](#); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);
26 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;
27 <https://www.drupal.org/docs/8/system-requirements/web-server>;
28 <https://www.drupal.org/docs/8/core/modules/rest/overview>;

1 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;
2 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;
3 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding
4 Magento, *see, e.g.*, [http://docs.magento.com/m1/ee/user_guide/system-operations/browser-](http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html)
5 [capabilities-detection.html](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html); [http://docs.magento.com/m1/ee/user_guide/system-operations/index-](http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html)
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-cms.html;
7 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-management.html. Regarding
17 Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;
18 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Hierarchy;
19 http://codex.wordpress.org/Function_Reference/the_title;
20 http://codex.wordpress.org/Function_Reference/the_content;
21 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
22 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
23 http://codex.wordpress.org/Query_Overview. Regarding Joomla, *see, e.g.*,
24 https://docs.joomla.org/Joomla_Browser_Support;
25 <https://www.joomlart.com/documentation/other/joomla-3-and-joomla-2-5-system-requirement>;
26 <https://showcase.joomla.org/>; https://docs.joomla.org/Editor_form_field_type;
27 <https://developer.joomla.org/coding-standards/html.html>; [https://developer.joomla.org/coding-](https://developer.joomla.org/coding-standards/css.html)
28 [standards/css.html](https://developer.joomla.org/coding-standards/javascript.html); <https://developer.joomla.org/coding-standards/javascript.html>;

1 https://docs.joomla.org/Generating_JSON_output; <https://api.joomla.org/cms->
2 [3/classes/Joomla.CMS.Input.Json.html](https://api.joomla.org/cms-3/classes/Joomla.CMS.Input.Json.html);
3 https://docs.joomla.org/How_do_you_assign_a_module_to_specific_pages%3F;
4 https://docs.joomla.org/Where_are_the_web_pages%3F.

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

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

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

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

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

28

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

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

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

1 46. The presence of the above referenced elements are demonstrated, by way of example,
2 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/project/ckeditor>;
3 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;
4 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
5 [images/10/03/2016/9821](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);
6 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
7 [images/10/03/2016/9821](https://www.drupal.org/docs/8/core/modules/media/overview); <https://www.drupal.org/docs/8/core/modules/media/overview>;
8 https://www.drupal.org/project/media_entity.

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

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

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

19 50. The presence of the above referenced elements are demonstrated, by way of example,
20 by reference to publicly available information. *See, e.g.*, <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/8/core/modules/custom_block/overview;
25 <https://www.drupal.org/docs/8/core/modules/contact/overview>.

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

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

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

7 54. The Accused Instrumentalities infringe claim 9 of the '397 patent through a
8 combination of features which collectively practice each limitation of claim 9. *See, e.g.*,
9 <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-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
12 [images/10/03/2016/9821](https://www.drupal.org/docs/8/core/modules/image/working-with-images); <https://www.drupal.org/docs/8/core/modules/image/working-with-images>;
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 <https://www.wpbeginner.com/wp-tutorials/how-to-add-custom-styles-to-wordpress-visual-editor/>.

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

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

26 57. Claim 11 recites the apparatus of claim 9, wherein said elements are described by a
27 transformation or a timelines of said selected styles.
28

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

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

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

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

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

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

21 64. By way of example, the Accused Instrumentalities enable descriptions of elements
22 describing CSS animations. *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)
23 [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>.

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

28

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

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

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

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

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

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

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

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

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

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

8 76. By way of example, the Accused Instrumentalities enable rescaling of a web page to
9 the size of the particular screen that is being used. *See, e.g.*,

10 <https://www.drupal.org/docs/8/mobile/responsive-web-design>;

11 <https://www.drupal.org/docs/8/mobile/web-based-mobile-apps>.

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

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

17 79. By way of example, the Accused Instrumentalities enable dynamic resizing upon
18 display to a different device and screen. For example, the Accused Instrumentalities include
19 "Responsive Web Design." Responsive Web Design refers to web design that changes formatting
20 and lay-out to respond to different devices, screen sizes and browser capabilities. The Accused
21 Instrumentalities therefore enable the creation of web pages that may be viewed with resizing means
22 operable to redefine a size of a web page upon being displayed. *See, e.g.*,

23 http://www.w3schools.com/html/html_responsive.asp;

24 <https://www.drupal.org/docs/8/mobile/responsive-web-design>;

25 <https://www.drupal.org/docs/8/mobile/web-based-mobile-apps>.

26 80. Claim 37 of the '397 patent generally recites [a]n apparatus for producing websites
27 with web page(s) on and for a computer with a browser and a virtual machine, the apparatus
28 comprising: an interface for building a website through control of website elements, being operable

1 through the browser on to: present a selectable settings menu, accept settings, and generate the
2 display in accordance with an assembly of settings contemporaneously with the acceptance thereof,
3 at least one setting being operable to generate said display through commands to said virtual
4 machine; an internal database associated with the interface for storing information representative of
5 one or more of assembly of settings for controlling elements of the website; and a build tool to
6 construct web page(s) of the website having: an external database containing data corresponding to
7 the information stored in the internal database, and one or more run time files, where said run time
8 files use information stored in the external database to generate virtual machine commands for the
9 display of at least a portion of one or more web pages.

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

25 82. By way of example, the Accused Instrumentalities include various multimedia objects
26 selected from a group contained within a WYSIWYG Editor. Examples include color, font, an
27 image, a video, a text area and a URL as they appear in the WYSIWYG Editor. The multimedia
28 objects created in the WYSIWYG editor are stored in the database and appear as HTML scripted

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

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

22 84. The presence of the above referenced elements are demonstrated, by way of example,
23 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/home>;
24 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;
25 <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-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)
28 [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);

1 https://www.drupal.org/project/save_draft; [https://www.drupal.org/docs/7/understanding-](https://www.drupal.org/docs/7/understanding-drupal/technology-stack)
2 [drupal/technology-stack](https://www.drupal.org/docs/8/system-requirements/web-server); <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 http://docs.magento.com/m1/ee/user_guide/system-operations/index-management.html;
8 http://docs.magento.com/m1/ee/user_guide/design/layout-updates.html;
9 [http://docs.magento.com/m1/ee/user_guide/system-](http://docs.magento.com/m1/ee/user_guide/system-operations/cache-page.html?Highlight=database%20retrieval)
10 [operations/media-storage.html](http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage.html); [http://docs.magento.com/m1/ee/user_guide/system-](http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage-database.html)
11 [operations/media-storage-database.html](http://docs.magento.com/m1/ee/user_guide/system-operations/media-storage-database.html);
12 [http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.p](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf)
13 [df](http://docs.magento.com/m1/ee/user_guide/store-operations/stores-multiple.html); http://docs.magento.com/m1/ee/user_guide/store-operations/stores-multiple.html;
14 http://docs.magento.com/m1/ee/user_guide/store-operations/store-hierarchy.html;
15 http://docs.magento.com/m1/ee/user_guide/system-operations/browser-capabilities-detection.html;
16 http://docs.magento.com/m1/ce/user_guide/design/page-layout.html;
17 http://docs.magento.com/m1/ce/user_guide/design/layout-updates.html; and
18 [http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.p](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf)
19 [df](http://docs.magento.com/m1/ee/user_guide/Resources/pdf/magento_enterprise_edition_user_guide.pdf). Regarding Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;
20 <http://codex.wordpress.org/Templates>; 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 http://codex.wordpress.org/Template_Tags/get_the_title;
24 http://codex.wordpress.org/Query_Overview; <https://www.wpbeginner.com/glossary/database/>; and
25 <https://codex.wordpress.org/Pages>. Regarding Joomla, *see, e.g.*,
26 https://docs.joomla.org/Joomla_Browser_Support;
27 <https://www.joomlart.com/documentation/other/joomla-3-and-joomla-2-5-system-requirement>;
28

1 <https://showcase.joomla.org/>; https://docs.joomla.org/Editor_form_field_type;
2 <https://developer.joomla.org/coding-standards/html.html>; <https://developer.joomla.org/coding-standards/css.html>; <https://developer.joomla.org/coding-standards/javascript.html>;
3 https://docs.joomla.org/Generating_JSON_output; <https://api.joomla.org/cms-3/classes/Joomla.CMS.Input.Json.html>;
4 https://docs.joomla.org/How_do_you_assign_a_module_to_specific_pages%3F;
5 https://docs.joomla.org/Where_are_the_web_pages%3F.

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

11 86. In particular, Defendant's actions that aid and abet others such as its partners,
12 customers, clients, and/or end users to infringe include advertising and distributing the Accused
13 Instrumentalities and providing instruction materials, training, and services regarding the Accused
14 Instrumentalities. *See, e.g.,* <https://blog.e-zest.com/have-you-tried-drupal-yet/>. On information and
15 belief, Defendant has engaged in such actions with specific intent to cause infringement or with
16 willful blindness to the resulting infringement because Defendant has had actual knowledge of the
17 '397 patent and knowledge that its acts were inducing infringement of the '397 patent since at least
18 the date e-Zest received notice that such activities infringed the '397 patent.

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

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

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

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

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

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

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

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

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

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

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

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

27
28

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

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

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

10 101. Upon information and belief, Defendant has and continues to directly infringe at least
11 claims 1-6 of the '168 patent by using a browser-based website and/or web page authoring tool in
12 which the user-selected settings representing website elements are stored in a database, and retrieval
13 of said information to generate said website (the "Accused Instrumentalities"). The Accused
14 Instrumentalities include but are not limited website building tools used and/or provided by
15 Defendant, such as, for example Drupal, Joomla, Magento and/or Wordpress. *See, e.g.*,
16 <https://www.e-zest.com/case-study-drupal-based-corporate-blogging-and-website>; <https://www.e-zest.com/joomla-development-services>; <https://www.e-zest.com/casestudy-enabling-unified-experience-with-headless-magento-and-aem-for-a-fortune-500-healthcare-giant>; <https://www.e-zest.com/case-study-transition-from-a-static-legacy-site-to-wordpress-cms>.

17
18
19
20 102. In particular, claim 1 of the '168 patent generally recites a system for assembling a
21 website comprising a server with a build engine, the website comprising web pages with objects (one
22 button or one image object), the server accepting user input to associate a style with objects, wherein
23 a button or image object is associated with a style that includes values defining transformations and
24 time lines; wherein each web page is defined entirely by the objects and the style associated with the
25 object, produce a database with a multidimensional array comprising the objects that comprise the
26 website including data defining the object style, number, and an indication of the web page that each
27 object is part of, and provide the database to a server accessible to web browser; wherein the
28

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

28

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

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

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

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

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

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

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

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

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

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

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

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

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

28

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

16 156. The Accused Instrumentalities infringe claim 2 of the '287 patent through a
17 combination of features which collectively practice each limitation of claim 1. By way of example,
18 the registry includes definitions of input and output related to a web service as evidenced by
19 WordPress's JSON formatting characteristics of the defined UI objects. (*See*, for example,
20 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
21 <https://developer.wordpress.org/rest-api/reference/posts/> https://developer.wordpress.org/rest-api/reference/posts/#schema-title_2, https://codex.wordpress.org/Widgets_API;
22 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
23 https://codex.wordpress.org/WordPress_Widgets_2.)

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

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

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

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

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

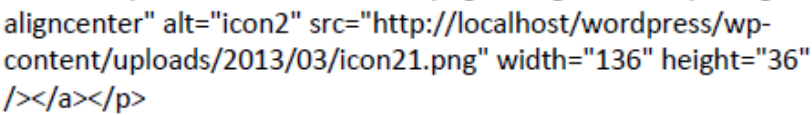
14 162. The Accused Instrumentalities infringe claim 5 of the '287 patent through a
15 combination of features which collectively practice each limitation of claim 1. By way of example,
16 the Accused Instrumentalities additionally feature a defined UI object that is an input field for a web
17 service.

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

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

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

The stored data in the wp_posts table for the “WordPress Info” web page includes information corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages 22%”. Other user selections shown in stored database data below include, for 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;">http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png</p> <p> 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, 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).

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

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

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

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

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

1 <https://torquemag.io/2017/08/make-wordpress-website-mobile-friendly/>;

2 <https://wordpress.org/themes/ultra/>.)

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

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

28

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

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

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

1 (See, for example, https://codex.wordpress.org/WordPress_Widgets,
2 https://codex.wordpress.org/Plugin_Resources, https://codex.wordpress.org/Plugin_API.)

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

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

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

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

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

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

27 181. The Accused Instrumentalities feature a system where the Player receives the output
28 symbolic name and corresponding one or more output values and provides instructions for the

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

7 182. The presence of the above referenced features is demonstrated, by way of example,
8 by reference to publicly available information. 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 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;
14 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
15 http://codex.wordpress.org/Query_Overview.

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

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

28

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

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

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

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

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

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

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

23 192. The Accused Instrumentalities infringe claim 25 of the '287 patent through a
24 combination of features which collectively practice each limitation of claim 25. WordPress sends all
25 files over a network using a variety of databases in its technology stack including MySQL. These
26 backend capabilities provided the code over a network. By way of example, data from the
27 wp_options table for the website header and from the wp_posts table for the "WordPress Info" web
28 page are extracted directly from the Bitnami WordPress server-side database using MySQL

1 Workbench. The stored data in the wp_options table includes the website's url, the website's title
2 (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

3
4
5
6
7
8 The stored data in the wp_posts table for the "WordPress Info" web page includes information
9 corresponding to user selected settings such as, for example, the color red ("#ff0000") for "manages
10 22%". Other user selections shown in stored database data below include, for ex-ample, the image
11 filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the
12 image alignment (class="wp-image-24 aligncenter"), and a selected paragraph style (h3) for
13 "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><h3>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</h3></p> <p>&nbsp;</p>	WordPress Info	page

24
25 WordPress's HTML, CSS, Java, and JSON coding capabilities further are shown, for example by
26 <http://codex.wordpress.org/Templates>, <https://codex.wordpress.org/CSS>,
27 <https://developer.wordpress.org/rest-api/> and https://codex.wordpress.org/Widgets_API.

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

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

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

10 196. In particular, Defendant's actions that aid and abet others such as its partners,
11 customers, clients, and/or end users to infringe include advertising and distributing the Accused
12 Instrumentalities and providing instruction materials, training, and services regarding the Accused
13 Instrumentalities. *See, e.g.*, <https://blog.e-zest.com/laravel-5-and-wordpress-integration>. On
14 information and belief, Defendant has engaged in such actions with specific intent to cause
15 infringement or with willful blindness to the resulting infringement because Defendant has had
16 actual knowledge of the '287 patent and knowledge that its acts were inducing infringement of the
17 '287 patent since at least the date e-Zest received notice that such activities infringed the '287 patent.

18 197. Upon information and belief, Defendant is liable as a contributory infringer of the
19 '287 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
20 website authoring tools to be especially made or adapted for use in an infringement of the '287
21 patent. The Accused Instrumentalities are a material component for use in practicing the '287 patent
22 and are specifically made and are not a staple article of commerce suitable for substantial non-
23 infringing use.

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

26 199. Since the date of the filing of this Complaint, Defendant's infringement of the '287
27 patent has been willful.
28

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

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

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

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

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

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

21 205. The claims of the '044 patent do not merely recite the performance of some business
22 practice known from the pre-Internet world along with the requirement to perform it on the Internet.
23 Instead, the claims of the '044 patent recite one or more inventive concepts that are rooted in the
24 computerized generation of content on a display of a device, such as a mobile device, and overcome
25 problems specifically arising in the realm of computerized display content generation technologies.

26 206. The claims of the '044 patent recite an invention that is not merely the routine or
27 conventional use of systems and methods for the computerized generation of content on a display of
28

1 a device. Instead, the invention describes systems for use with devices with authoring tools or
2 Players specific to each device and Applications that are device independent.

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

7 208. Accordingly, each claim of the '044 patent recites a combination of elements
8 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an
9 ineligible concept.

10 209. Plaintiff is the assignee and owner of the right, title and interest in and to the '044
11 patent, including the right to assert all causes of action arising under the patents and the right to any
12 remedies for infringement of them.

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

24 211. In particular, claim 1 of the '044 patent recites a system for generating code to
25 provide content on a display of a device, the system comprising: computer memory storing: a)
26 symbolic names required for evoking one or more web components each related to a set of inputs
27 and outputs of a web service obtainable over a network, where the symbolic names are character
28 strings that do not contain either a persistent address or pointer to an output value accessible to the

1 web service, where each symbolic name has an associated data format class type corresponding to a
2 subclass of User Interface (UI) objects that support the data format type of the symbolic name, and
3 where each symbolic name has a preferred UI object, and b) an address of the web service; an
4 authoring tool configured to: define a UI object for presentation on the display, where the defined UI
5 object corresponds to a web component included in the computer memory selected from a group
6 consisting of an input of the web service and an output of the web service, where each defined UI
7 object is either: 1) selected by a user of the authoring tool; or 2) automatically selected by the system
8 as the preferred UI object corresponding to the symbolic name of the web component selected by the
9 user of the authoring tool, access the computer memory to select the symbolic name corresponding
10 to the web component of the defined UI object, associate the selected symbolic name with the
11 defined UI object, where the selected symbolic name is only available to UI objects that support the
12 defined data format associated with that symbolic name, store information representative of the
13 defined UI object and related settings in a database; retrieve the information representative of the
14 one or more the UI object settings stored in the database; and build an Application consisting of one
15 or more web page views from at least a portion of the database utilizing at least one Player, where
16 the Player utilizes information stored in the database to generate for the display of at least a portion
17 of the one or more web pages, wherein when the Application and Player are provided to the device
18 and executed on the device, and when the user of the device provides one or more input values
19 associated with an input symbolic name to an input of the defined UI object, the device provides the
20 user provided one or more input values and corresponding input symbolic name to the web service,
21 the web service utilizes the input symbolic name and the user provided one or more input values for
22 generating one or more output values having an associated output symbolic name, and the Player
23 receives the output symbolic name and corresponding one or more output values and provides
24 instructions for the display of the device to present an output value in the defined UI object.

25 212. The Accused Instrumentalities infringe claim 1 of the '044 patent through a
26 combination of features which collectively practice each limitation of claim 1. By way of example,
27 the Accused Instrumentalities feature a system for generating code to provide content on a display of
28 a device. The system includes a WordPress server, which provides WordPress's WYSIWYG visual

1 effects editor and a WordPress-compatible browser. WordPress’s WYSIWYG visual effects editor
 2 generates code, such as JavaScript or HTML code for such as options for defining title, text, images,
 3 videos and paragraph styles, while the browser displays the resulting content as a WordPress
 4 webpage on a display of a device, such as a computer display.

5 213. For example, on information and belief, WordPress uses a variety of databases in its
 6 technology stack including MySQL. Data from the wp_options table for the website header and the
 7 data from the wp_posts table for the “WordPress Info” web page extracted directly from the Bitnami
 8 WordPress server-side database using MySQL Workbench. The stored data in the wp_options table
 9 includes the website’s url, the website’s title (blogname), the website’s tagline (blogdescription), and
 10 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

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

<http://codex.wordpress.org/Templates>, <https://codex.wordpress.org/CSS>,

<https://developer.wordpress.org/rest-api/>.

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

215. The computer memory stores a) symbolic names required for evoking one or more web components each related to a set of inputs and outputs of a web service obtainable over a network, where the symbolic names are character strings that do not contain either a persistent address or pointer to an output value accessible to the web service. The WordPress MySQL database contains symbolic names required for evoking one or more web components each related to a set of inputs and outputs of a web service obtainable over a network as demonstrated by the formatting in conjunction with WordPress's WYSIWYG visual effects editor, widget, and plugin authoring tools. JSON names are strings that only represent the symbolic names that are bound both to a web service input and/or output and to a UI object. All JSON names in the name/value pairs are character strings. WordPress' WYSIWYG visual effects editor includes elements for defining the

1 layout for placement of the defined UI objects. Widgets, plug-ins and other elements correspond to
2 the defined UI objects and are the product of the JSON formatting. (*See*, for example,
3 https://codex.wordpress.org/WordPress_Lessons#Template_Files,
4 https://codex.wordpress.org/WordPress_Widgets, https://codex.wordpress.org/Plugin_Resources,
5 and https://codex.wordpress.org/Plugin_API.)

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

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

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

27 219. The authoring tool is configured to define a UI object for presentation on the display,
28 where the defined UI object corresponds to a web component included in the computer memory

1 selected from a group consisting of an input of the web service and an output of the web service.
2 WordPress's WYSIWYG visual effects editor and widget authoring tools define the presence of a UI
3 object for presentation on a display and the defined UI object corresponds to a web component
4 included in the computer memory selected from a group consisting of an input of the web service
5 and an output of the web service.

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

16 221. The authoring tool is configured to access the computer memory to select the
17 symbolic name corresponding to the web component of the defined UI object based on its JSON
18 formatting characteristics.

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

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

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

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

21 226. The Accused Instrumentalities feature a system where the Application and Player are
22 provided to the device and executed on the device.

23 227. When the user of the device provides one or more input values associated with an
24 input symbolic name to an input of the defined UI object the device provides the user provided one
25 or more input values and corresponding input symbolic name to the web service. Because the
26 Accused Instrumentalities incorporate a system that includes WordPress, when a user of the device
27 provides one or more input values associated with an input symbolic name, using JSON formatting
28 characteristics, to an input of the defined UI object, the device provides the user provided one or

1 more input values and corresponding input symbolic name, using JSON formatting characteristics, to
2 the web service. (*See*, for example, <https://developer.wordpress.org/rest-api/>,
3 <https://developer.wordpress.org/rest-api/reference/>, https://codex.wordpress.org/Widgets_API,
4 https://codex.wordpress.org/Plugin_Resources, and https://codex.wordpress.org/Plugin_API.)

5 228. The Accused Instrumentalities feature a system where the web service utilizes the
6 input symbolic name and the user provided one or more input values for generating one or more
7 output values having an associated output symbolic name. Because of the JSON formatting, the
8 output values having an associated output symbolic name. (*See*, for example,
9 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,
10 https://codex.wordpress.org/Widgets_API, https://codex.wordpress.org/Plugin_Resources, and
11 https://codex.wordpress.org/Plugin_API.)

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

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

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

5 232. The Accused Instrumentalities infringe claim 2 of the '044 patent through a
6 combination of features which collectively practice each limitation of claim 2. By way of example,
7 the registry includes definitions of input and output related to a web service as evidenced by
8 WordPress's JSON formatting characteristics of the defined UI objects. (*See*, for example,
9 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
10 <https://developer.wordpress.org/rest-api/reference/posts/> <https://developer.wordpress.org/rest-api/reference/posts/#schema-title> https://codex.wordpress.org/Widgets_API;
11 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
12 https://codex.wordpress.org/WordPress_Widgets.)

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

16 234. The Accused Instrumentalities infringe claim 3 of the '044 patent through a
17 combination of features which collectively practice each limitation of claim 3. By way of example,
18 the Accused Instrumentalities feature web components additionally including web chat, Reuters
19 RSS feed, Calendar image, and map image widgets.

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

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

26 237. Claim 5 of the '044 patent recites a system for generating code to provide content on
27 a display of a device that includes all the elements of claim 1, additionally where the system stores
28

1 information in a registry, and wherein the registry includes definitions of input and output related to
2 the web service.

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

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

10 240. The Accused Instrumentalities infringe claim 11 of the '044 patent through a
11 combination of features which collectively practice each limitation of claim 11. WordPress sends all
12 files over a network using a variety of databases in its technology stack including MySQL. These
13 backend capabilities provided the code over a network. By way of example, data from the
14 wp_options table for the website header and from the wp_posts table for the “WordPress Info” web
15 page are extracted directly from the Bitnami WordPress server-side database using MySQL
16 Workbench. The stored data in the wp_options table includes the website’s url, the website’s title
17 (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

18
19
20
21
22
23 The stored data in the wp_posts table for the “WordPress Info” web page includes information
24 corresponding to user selected settings such as, for example, the color red (“#ff0000”) for “manages
25 22%”. Other user selections shown in stored database data below include, for ex-ample, the image
26 filename for the image (<http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png>), the
27 image alignment (class=“wp-image-24 aligncenter”), and a selected paragraph style (h3) for
28 “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>&nbsp;</p> <p><h3>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</h3></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).

241. 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.

242. 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>.

243. 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 on the device. The non-volatile computer memory stores symbolic names required for evoking one or more web components each related to a set of inputs and outputs

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

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

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

18 246. The Accused Instrumentalities include defining a UI object for presentation on the
19 display, where the UI object corresponds to a web component included in the computer memory,
20 where the web component is selected from a group consisting of an input of a web service and an
21 output of the web service. WordPress defines a user interface object, *i.e.*, an element/UI component,
22 for presentation on display, where the UI object corresponds to a web component included in the
23 non-volatile computer memory selected from a group consisting of an input of a web service and an
24 output of the web service (as evidenced by JSON data formatting)..

25 247. Each defined UI object is either: 1) selected by a user of an authoring tool; or 2)
26 automatically selected by the system as the preferred UI object corresponding to the symbolic name
27 of the web component selected by the user of the authoring tool. When a Widget is selected in the
28 WordPress Widget selection list (See https://codex.wordpress.org/WordPress_Widgets) steps 1 to 5

1 under “Displaying Widgets“, the widget UI will automatically displayed in the Web Page Sidebar.
2 WordPress contains an authoring tool in the form of the WYSIWYG visual effects editor, widgets,
3 and plug-in authoring tools.

4 248. The Accused Instrumentalities include selecting the symbolic name corresponding to
5 the web component (i.e. WordPress Widget or Plug-in) of the defined UI object and associating the
6 selected symbolic name with the defined UI object, where the selected symbolic name is only
7 available to UI objects that support the defined data format associated with that symbolic name.
8 WordPress accesses the non-volatile memory to select the symbolic name corresponding to the web
9 component of the defined UI object (as evidenced by JSON data formatting), associate the selected
10 symbolic name with the defined UI object (the JSON element corresponding to an element), where
11 the selected symbolic name is only available to UI objects that support the defined data format
12 associated with that symbolic name (the element associated with at JSON string).

13 249. The Accused Instrumentalities also include retrieving the information representative
14 of the one or more the UI object settings stored in the database based on the JSON formatting
15 characteristics.

16 250. The Accused Instrumentalities include building an Application consisting of one or
17 more web page views from at least a portion of the database utilizing the Player, where the Player
18 utilizes information stored in the database to generate for the display of at least a portion of the one
19 or more web pages. WordPress builds an Application included in the symbolic name of the defined
20 UI object.

21 251. With the Accused Instrumentalities when the Application and Player are provided to
22 the device and executed on the device when the Application and Player are provided to the device
23 and executed on the device, and when the user of the device provides one or more input values
24 associated with an input symbolic name to an input of the defined UI object, 1) the device provides
25 the user provided one or more input values and corresponding input symbolic name to the web
26 service, 2) the web service utilizes the input symbolic name and the user provided one or more input
27 values for generating one or more output values having an associated output symbolic name, and 3)
28 the Player receives the output symbolic name and corresponding one or more output values and

1 provides instructions for the display of the device to present an output value in the defined UI object.
2 For example, in WordPress, a user of a device provides an input value associated with an input
3 symbolic name to an input of a defined UI object, such as utilizing an element, plug-in, or widget.
4 The element, plug-in, or widget is associated with symbolic name based on their JSON formatting
5 characteristics. 1) The element input value corresponds to the input symbolic name based on its
6 JSON formatting characteristics 2) The element/UI component output value corresponds to the
7 output symbolic name via JSON. 3) The runtime player within WordPress receives the output name,
8 output value, and provides instructions for a display as shown by the fact that the defied UI object is
9 displayed.

10 252. The presence of the above referenced features is demonstrated, by way of example,
11 by reference to publicly available information. Regarding WordPress, see, e.g.,
12 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;
13 http://codex.wordpress.org/Template_Hierarchy;
14 http://codex.wordpress.org/Function_Reference/the_title;
15 http://codex.wordpress.org/Function_Reference/the_content;
16 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages1>
17 <http://codex.wordpress.org/Templates>; http://codex.wordpress.org/Template_Tags/get_the_title; and
18 http://codex.wordpress.org/Query_Overview.

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

22 254. The Accused Instrumentalities infringe claim 16 of the '044 patent through a
23 combination of features which collectively practice each limitation of claim 16. By way of example,
24 the Accused Instrumentalities feature storing information in a WordPress registry that includes
25 definitions of input and output related to a web service as evidenced by WordPress's JSON
26 formatting characteristics and the UI object functionality. See, e.g.,
27 <https://developer.wordpress.org/rest-api/>; <https://developer.wordpress.org/rest-api/reference/>;
28 <https://developer.wordpress.org/rest-api/reference/posts/> <https://developer.wordpress.org/rest->

1 [api/reference/posts/#schema-title https://codex.wordpress.org/Widgets_API](https://codex.wordpress.org/Widgets_API);
2 https://codex.wordpress.org/Plugin_Resources; and https://codex.wordpress.org/Plugin_API;
3 https://codex.wordpress.org/WordPress_Widgets.

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

7 256. The Accused Instrumentalities infringe claim 17 of the '044 patent through a
8 combination of features which collectively practice each limitation of claim 17. By way of example,
9 the Accused Instrumentalities feature web components additionally including web chat, Reuters RSS
10 feed, Calendar image, and map image widgets.

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

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

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

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

22 261. Claim 25 of the '044 patent recites a method of displaying content on a display that
23 includes all the elements of claim 15, additionally where the method includes providing the code over
24 the network.

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

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

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

10 265. In particular, Defendant's actions that aid and abet others such as its partners,
11 customers, clients, and/or end users to infringe include advertising and distributing the Accused
12 Instrumentalities and providing instruction materials, training, and services regarding the Accused
13 Instrumentalities. See, e.g., <https://blog.e-zest.com/laravel-5-and-wordpress-integration>. On
14 information and belief, Defendant has engaged in such actions with specific intent to cause
15 infringement or with willful blindness to the resulting infringement because Defendant has had
16 actual knowledge of the '044 patent and knowledge that its acts were inducing infringement of the
17 '044 patent since at least the date e-Zest received notice that such activities infringed the '044 patent.

18 266. Upon information and belief, Defendant is liable as a contributory infringer of the
19 '044 patent under 35 U.S.C. § 271(c) by offering to sell, selling and importing into the United States
20 website authoring tools to be especially made or adapted for use in an infringement of the '044
21 patent. The Accused Instrumentalities are a material component for use in practicing the '044 patent
22 and are specifically made and are not a staple article of commerce suitable for substantial non-
23 infringing use.

24 267. Defendant was made aware of the '044 patent and its infringement thereof at least as
25 early as the filing of this Complaint.

26 268. Since the date of the filing of this Complaint, Defendant's infringement of the '044
27 patent has been willful.
28

