

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  
6 DEVLIN LAW FIRM LLC  
2069 Cold Canyon Road  
7 Calabasas, CA 91302

8 Robert Kiddie (pro hac vice to be filed)  
rkiddie@devlinlawfirm.com  
9 DEVLIN LAW FIRM LLC  
1306 N. Broom St., 1st Floor  
10 Wilmington, DE 19806  
Telephone: (302) 449-9010  
11 Facsimile: (302) 353-4251

12 Attorneys for Express Mobile, Inc.

13  
14 UNITED STATES DISTRICT COURT  
15 NORTHERN DISTRICT OF CALIFORNIA  
16

17 EXPRESS MOBILE, INC., ) Case No.: 3:18-cv-04688-RS  
18 Plaintiff, )  
19 vs. ) **FIRST AMENDED COMPLAINT FOR**  
20 ) **PATENT INFRINGEMENT**  
21 PANTHEON SYSTEMS INC., ) DEMAND FOR JURY TRIAL  
22 )  
23 )  
24 Defendant. \_\_\_\_\_

1 Plaintiff Express Mobile, Inc. (“Express Mobile” or “Plaintiff”), for its First Amended  
2 Complaint against Defendant Pantheon Systems Inc., (“Pantheon” or “Defendant”) alleges the  
3 following:

4 **NATURE OF THE ACTION**

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

7 **THE PARTIES**

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

10 3. Upon information and belief, Pantheon is a corporation organized and existing under the  
11 laws of Delaware, with a place of business at 717 California Street, 3<sup>rd</sup> Floor, San Francisco, CA  
12 94108 and can be served through its registered agent, The Corporation Trust Company, Corporation  
13 Trust Center, 1209 Orange Street, Wilmington, DE 19801.

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

18 **JURISDICTION AND VENUE**

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

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

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

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

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

5 **COUNT I – INFRINGEMENT OF U.S. PATENT NO. 6,546,397**

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

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

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

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

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

27  
28

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

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

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

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

23           18.     Plaintiff is the assignee and owner of the right, title and interest in and to the '397  
24 patent, including the right to assert all causes of action arising under said patents and the right to any  
25 remedies for infringement of them.

26           19.     Upon information and belief, Defendant has and continues to directly infringe at least  
27 claims 1-6, 8-11, 14-15, 24-25, 35, and 37 of the '397 patent by using a browser-based website  
28 and/or web page authoring tool in which the user-selected settings representing website elements are

1 stored in a database, and in which said stored information is retrieved to generate said website (the  
2 “Accused Instrumentalities”). The Accused Instrumentalities include but are not limited to the  
3 website building tools used and/or provided by Defendant, such as, for example Drupal and/or  
4 Wordpress. *See, e.g.*, <https://pantheon.io/features/drupal-hosting>;  
5 <https://pantheon.io/features/wordpress-hosting-on-pantheon?action>.

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

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

23 22. The Accused Instrumentalities infringe claim 1 of the ’397 patent through a  
24 combination of features which collectively practice each limitation of claim 1. 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://themeforest.net/category/wordpress>.) All of these browsers rely on browser engines  
8 comprising virtual machines to interpret and execute JavaScript and HTML to render web pages on a  
9 computer.

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

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

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

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

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

16 27. The presence of the above referenced elements are demonstrated, by way of example,  
17 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/home>;  
18 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;  
19 <https://www.drupal.org/project/ckeditor>;  
20 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;  
21 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)  
22 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);  
23 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;  
24 <https://www.drupal.org/docs/8/system-requirements/web-server>;  
25 <https://www.drupal.org/docs/8/core/modules/rest/overview>;  
26 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;  
27 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;  
28 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding

1 Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;  
2 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy);  
3 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
4 [http://codex.wordpress.org/Function\\_Reference/the\\_content](http://codex.wordpress.org/Function_Reference/the_content);  
5 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;  
6 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Tags/get\\_the\\_title](http://codex.wordpress.org/Template_Tags/get_the_title); and  
7 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview).

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

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



1 comprising virtual machines to interpret and execute JavaScript and HTML to render web pages on a  
2 computer.

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

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

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

1 a portion of the page directly, or generate HTML to be executed for display by the main layout  
2 engine.

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

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

10 35. The presence of the above referenced elements are demonstrated, by way of example,  
11 by reference to publicly available information. Regarding Drupal, *see, e.g.*,  
12 <https://www.drupal.org/home>; [https://www.drupal.org/docs/8/system-requirements/browser-](https://www.drupal.org/docs/8/system-requirements/browser-requirements)  
13 [requirements; https://www.drupal.org/project/ckeditor](https://www.drupal.org/project/ckeditor);  
14 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;  
15 [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)  
16 [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);  
17 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;  
18 <https://www.drupal.org/docs/8/system-requirements/web-server>;  
19 <https://www.drupal.org/docs/8/core/modules/rest/overview>;  
20 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;  
21 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;  
22 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding  
23 Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;  
24 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy);  
25 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
26 [http://codex.wordpress.org/Function\\_Reference/the\\_content](http://codex.wordpress.org/Function_Reference/the_content);  
27 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;

28

1 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Tags/get\\_the\\_title](http://codex.wordpress.org/Template_Tags/get_the_title); and  
2 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview).

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

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

8 38. By way of example, the JSON strings that are used to generate, in part, field  
9 capabilities originate from the database and therefore reflect the database structure and contents  
10 showing, on information and belief, the implementation of a multidimensional array structured  
11 database. By way of further evidence, the JSON strings show that there are dimensions for various  
12 parameters. Regarding Drupal, *see, e.g.*, <https://www.drupal.org/files/issues/Field.png>;  
13 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>.  
14 Regarding Wordpress, *see, e.g.*, <https://code.tutsplus.com/>;  
15 <https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure>.  
16 structure.

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

20 40. The Accused Instrumentalities infringe claim 4 of the '397 patent through a  
21 combination of features that practice the limitations of Claim 4. Regarding Drupal, *see, e.g.*,  
22 <https://www.drupal.org/docs/8/api/entity-api/defining-and-using-content-entity-field-definitions>.  
23 Regarding Wordpress, *see, e.g.*, [https://wordpress.stackexchange.com/questions/43302/wordpress-](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure)  
24 [settings-api-and-option-array-structure](https://wordpress.stackexchange.com/questions/43302/wordpress-settings-api-and-option-array-structure).

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

28

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

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

25           44.     The presence of the above referenced elements are demonstrated, by way of example,  
26 by reference to publicly available information. *See, e.g.*, <https://www.drupal.org/project/ckeditor>;  
27 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;  
28 <https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive->

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

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

8 46. The Accused Instrumentalities infringe claim 6 of the '397 patent through a  
9 combination of features that practice the limitations of Claim 6.

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

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

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

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

28

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

4           52.     The Accused Instrumentalities infringe claim 9 of the '397 patent through a  
5 combination of features which collectively practice each limitation of claim 9. Regarding Drupal  
6 *see, e.g.*, <https://www.drupal.org/project/ckeditor>;  
7 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;  
8 [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)  
9 [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>;  
10 <https://www.drupal.org/docs/7/understanding-drupal/technology-stack>;  
11 <https://www.drupal.org/docs/8/system-requirements/web-server>;  
12 <https://www.drupal.org/docs/8/core/modules/rest/overview>;  
13 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;  
14 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;  
15 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding  
16 Wordpress, *see, e.g.*, [https://www.wpbeginner.com/wp-tutorials/how-to-add-custom-styles-to-](https://www.wpbeginner.com/wp-tutorials/how-to-add-custom-styles-to-wordpress-visual-editor/)  
17 [wordpress-visual-editor/](https://www.wpbeginner.com/wp-tutorials/how-to-add-custom-styles-to-wordpress-visual-editor/).

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

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

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

27           56.     The Accused Instrumentalities infringe claim 11 of the '397 patent through a  
28 combination of features which collectively practice each limitation of claim 11. By way of example,

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

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

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

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

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

17 61. The Accused Instrumentalities infringe claim 15 of the '397 patent through a  
18 combination of features which collectively practice each limitation of claim 15.

19 62. By way of example, the Accused Instrumentalities enable descriptions of elements  
20 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)  
21 [architecture-for-drupal-8](http://demos.dojotoolkit.org/demos/css3/demo.html); <http://demos.dojotoolkit.org/demos/css3/demo.html>.

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

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

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

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

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

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

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

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

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

21         71. By way of example, the Accused Instrumentalities enable dynamic resizing upon  
22 display to a different device and screen. For example, the Accused Instrumentalities include  
23 “Responsive Web Design.” Responsive Web Design refers to web design that changes formatting  
24 and lay-out to respond to different devices, screen sizes and browser capabilities. The Accused  
25 Instrumentalities therefore enable the creation of web pages that may be viewed with resizing means  
26 operable to redefine a size of a web page upon being displayed. *See, e.g.*,  
27 [http://www.w3schools.com/html/html\\_responsive.asp](http://www.w3schools.com/html/html_responsive.asp);

28



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

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

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

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

28

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

24           75. Furthermore, the Accused Instrumentalities enable data from the client-side form  
25 referenced to be stored in a server-side database.

26           76. The presence of the above referenced elements are demonstrated, by way of example,  
27 by reference to publicly available information. Regarding Drupal, *see, e.g.*,  
28 <https://www.drupal.org/home>; <https://www.drupal.org/docs/8/system-requirements/browser->

1 requirements; <https://www.drupal.org/project/ckeditor>;  
2 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;  
3 [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)  
4 [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);  
5 [https://www.drupal.org/project/save\\_draft](https://www.drupal.org/project/save_draft); [https://www.drupal.org/docs/7/understanding-](https://www.drupal.org/docs/7/understanding-drupal/technology-stack)  
6 [drupal/technology-stack](https://www.drupal.org/docs/7/understanding-drupal/technology-stack); <https://www.drupal.org/docs/8/system-requirements/web-server>;  
7 <https://www.drupal.org/docs/8/core/modules/rest/overview>;  
8 <https://www.drupal.org/docs/8/core/modules/serialization/overview>;  
9 <https://www.drupal.org/docs/8/understanding-drupal-8/overview>;  
10 <https://www.drupal.org/docs/develop/standards/css/css-architecture-for-drupal-8>. Regarding  
11 Wordpress, *see, e.g.*, <http://themeforest.net/category/wordpress>;  
12 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy);  
13 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
14 [http://codex.wordpress.org/Function\\_Reference/the\\_content](http://codex.wordpress.org/Function_Reference/the_content);  
15 [http://codex.wordpress.org/Template\\_Tags/get\\_the\\_title](http://codex.wordpress.org/Template_Tags/get_the_title);  
16 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview); <https://www.wpbeginner.com/glossary/database/>; and  
17 <https://codex.wordpress.org/Pages>.

18 77. Defendant was made aware of the '397 patent and its infringement thereof at least as  
19 early as the filing of this Complaint.

20 78. Since the date of the filing of this Complaint, Defendant's infringement of the '397  
21 patent has been willful.

22 79. Plaintiff has been harmed by Defendant's infringing activities.

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

24 80. The allegations set forth in the foregoing paragraphs 1 through 79 are incorporated  
25 into this Second Claim for Relief.

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

1           82.     The inventions of the '168 patent resolve technical problems related to website  
2 creation and generation. For example, the inventions enable the creation of websites through  
3 browser-based build tools and a user interface, which features are exclusively implemented utilizing  
4 computer technology.

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

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

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

17          86.     Accordingly, each claim of the '168 patent recites a combination of elements  
18 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an  
19 ineligible concept.

20          87.     As noted above and incorporated into this Second Claim for Relief, a defendant in  
21 another case in which the '397 and '168 patents were asserted, asserted that the '397 and '168  
22 patents were invalid under 35 U.S.C. § 101. That motion and related Order are discussed above.

23          88.     Plaintiff is the assignee and owner of the right, title and interest in and to the '168  
24 patent, including the right to assert all causes of action arising under said patents and the right to any  
25 remedies for infringement of them.

26          89.     Upon information and belief, Defendant has and continues to directly infringe at least  
27 claims 1, 4, and 6 of the '168 patent by using a browser-based website and/or web page authoring  
28 tool in which the user-selected settings representing website elements are stored in a database, and

1 retrieval of said information to generate said website (the “Accused Instrumentalities”). The  
2 Accused Instrumentalities include but are not limited website building tools used and/or provided by  
3 Defendant, such as, for example Drupal. *See, e.g.*, <https://pantheon.io/features/drupal-hosting>.

4 90. In particular, claim 1 of the ’168 patent generally recites a system for assembling a  
5 website comprising a server with a build engine, the website comprising web pages with objects (one  
6 button or one image object), the server accepting user input to associate a style with objects, wherein  
7 a button or image object is associated with a style that includes values defining transformations and  
8 time lines; wherein each web page is defined entirely by the objects and the style associated with the  
9 object, produce a database with a multidimensional array comprising the objects that comprise the  
10 website including data defining the object style, number, and an indication of the web page that each  
11 object is part of, and provide the database to a server accessible to web browser; wherein the  
12 database is produced such that a web browser with access to a runtime engine is configured to  
13 generate the website from the objects and style data extracted from the provided database.

14 91. The Accused Instrumentalities infringe claim 1 of the ’168 patent through a  
15 combination of features which collectively practice each limitation of claim 1.

16 92. Further, by way of example, the JSON strings that are used by the Accused  
17 Instrumentalities to generate, in part, element formatting originate from the database and therefore  
18 reflect the database structure and contents showing, on information and belief, the implementation of  
19 a multidimensional array structured database comprising the objects that comprise the web site. By  
20 way of further evidence, the JSON strings show that there are dimensions for the pages, for arrays of  
21 columns, for arrays of sections, and for arrays of modules generated using the Accused  
22 Instrumentalities. *See, e.g.*, <https://www.drupal.org/files/issues/Field.png>;

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

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

1           94. By way of example, for the completed web site, the Accused Instrumentalities include  
2 runtime files, such as, for example HTML CSS files. *See, e.g.*, <https://www.drupal.org/home>;  
3 <https://www.drupal.org/docs/8/system-requirements/browser-requirements>;  
4 <https://www.drupal.org/project/ckeditor>;  
5 <https://www.drupal.org/docs/8/core/modules/ckeditor/overview>;  
6 [https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821)  
7 [images/10/03/2016/9821](https://dev.acquia.com/blog/tutorial-drupal-8-wysiwyg-inline-and-responsive-images/10/03/2016/9821); Angela Byron, *Ultimate Guide to Drupal 8* at 4 (2016);  
8 <https://www.drupal.org/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>;  
14 <https://www.drupal.org/docs/8/core/modules/media/overview>;  
15 [https://www.drupal.org/project/media\\_entity](https://www.drupal.org/project/media_entity);  
16 <https://www.drupal.org/docs/8/core/modules/image/working-with-images>;  
17 <http://demos.dojotoolkit.org/demos/css3/demo.html>; <https://www.drupal.org/files/issues/Field.png>;  
18 <https://api.drupal.org/api/drupal/core%21modules%21field%21field.module/group/field/8.3.x>.;  
19 [https://www.drupal.org/project/save\\_draft](https://www.drupal.org/project/save_draft).

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

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

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

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

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

6           100.    Defendant was made aware of the '168 patent and its infringement thereof at least as  
7 early as the filing of this Complaint.

8           101.    Since the date of the filing of this Complaint, Defendant's infringement of the '168  
9 patent has been willful.

10          102.    Plaintiff has been harmed by Defendant's infringing activities.

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

12          103.    The allegations set forth in the foregoing paragraphs 1 through 102 are incorporated  
13 into this Third Claim for Relief.

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

18          105.    The inventions of the '287 patent resolve technical problems related to generating  
19 content on a display of a device, such as the display of a mobile device. For example, the inventions  
20 of the '287 patent feature a registry and an authoring tool or Player configured to define a User  
21 Interface ("UI") object for display on the device, where the UI object corresponds to a web  
22 component. Each UI object is either: 1) selected by a user or 2) automatically selected by the system  
23 as a preferred UI object corresponding to a symbolic name of the web component and used to  
24 produce an Application, where the Application is a device-independent code; and a Player, where the  
25 Player is a device-dependent code. The Application and Player enable 1) the device to provide one  
26 or more input values and corresponding input symbolic name to the web service and 2) the web  
27 service to utilize the input symbolic name and the user provided one or more input values to generate  
28 one or more output values having an associated output symbolic name, while 3) the Player receives

1 the output symbolic name and corresponding one or more output values and provide instructions for  
2 the display of the device to present an output value in the defined UI object. These features are  
3 exclusively implemented utilizing computer technology.

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

9 107. The claims of the '287 patent recite inventions that are not merely the routine or  
10 conventional use of systems and methods for the computerized generation of content on a display of  
11 a device. Instead, the inventions feature systems for use with devices and methods of using the  
12 systems with authoring tools or Players specific to each device and Applications that are device  
13 independent.

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

18 109. Accordingly, each claim of the '287 patent recites a combination of elements  
19 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an  
20 ineligible concept.

21 110. Plaintiff is the assignee and owner of the right, title and interest in and to the '287  
22 patent, including the right to assert all causes of action arising under the patents and the right to any  
23 remedies for infringement of them.

24 111. Upon information and belief, Defendant has and continues to directly infringe at least  
25 claims 1-5, 11, 12, 15-19, 25 and 26 of the '287 patent by a system and method which includes a  
26 registry and an authoring tool or Player configured to define a User Interface ("UI") object for  
27 display on the device, where the UI object corresponds to a web component. Each UI object is  
28 either: 1) selected by a user or 2) automatically selected by the system as a preferred UI object



1 corresponding to a symbolic name of the web component and used to produce an Application, where  
2 the Application is a device-independent code and a Player, where the Player is a device-dependent  
3 code. The Application and Player enable 1) the device to provide one or more input values and  
4 corresponding input symbolic name to the web service and 2) the web service to utilize the input  
5 symbolic name and the user provided one or more input values to generate one or more output  
6 values having an associated output symbolic name, while 3) the Player receives the output symbolic  
7 name and corresponding one or more output values and provides instructions for the display of the  
8 device to present an output value in the defined UI object (the “Accused Instrumentalities”). The  
9 Accused Instrumentalities include platforms that enable the functionality described above and  
10 include but are not limited to, for example, WordPress. See, e.g.,  
11 <https://pantheon.io/product/wordpress-hosting?action=>. For example, Defendant’s managed  
12 WordPress hosting services handles the architecture, capacity, setup, and more for its users’  
13 WordPress websites.

14 112. In particular, claim 1 of the ’287 patent recites 1 a system for generating code to  
15 provide content on a display of a device, the system comprising: computer memory storing a registry  
16 of: a) symbolic names required for evoking one or more web components each related to a set of  
17 inputs and outputs of a web service obtainable over a network, where the symbolic names are  
18 character strings that do not contain either a persistent address or pointer to an output value  
19 accessible to the web service, where each symbolic name has an associated data format class type  
20 corresponding to a subclass of User Interface (UI) objects that support the data format type of the  
21 symbolic name, and has a preferred UI object, and b) an address of the web service; an authoring  
22 tool configured to: define a (UI) object for presentation on the display, where the defined UI object  
23 corresponds to a web component included in the registry selected from a group consisting of an input  
24 of the web service and an output of the web service, where each defined UI object is either: 1)  
25 selected by a user of the authoring tool; or 2) automatically selected by the system as the preferred  
26 UI object corresponding to the symbolic name of the web component selected by the user of the  
27 authoring tool, access the computer memory to select the symbolic name corresponding to the web  
28 component of the defined UI object, associate the selected symbolic name with the defined UI

1 object, where the selected symbolic name is only available to UI objects that support the defined  
2 data format associated with that symbolic name, and produce an Application including the selected  
3 symbolic name of the defined UI object, where the Application is a device-independent code; and a  
4 Player, where the Player is a device-dependent code, wherein, when the Application and Player are  
5 provided to the device and executed on the device, and when the user of the device provides one or  
6 more input values associated with an input symbolic name to an input of the defined UI object, 1)  
7 the device provides the user provided one or more input values and corresponding input symbolic  
8 name to the web service, 2) the web service utilizes the input symbolic name and the user provided  
9 one or more input values for generating one or more output values having an associated output  
10 symbolic name, 3) the Player receives the output symbolic name and corresponding one or more  
11 output values and provides instructions for the display of the device to present an output value in the  
12 defined UI object.

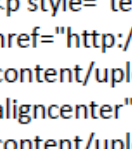
13 113. The Accused Instrumentalities infringe claim 1 of the '287 patent through a  
14 combination of features The Accused Instrumentalities infringe claim 1 of the '287 patent through a  
15 combination of features which collectively practice each limitation of claim 1. By way of example,  
16 the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

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

option_id	option_name	option_value
1	siteurl	<a href="http://localhost/wordpress">http://localhost/wordpress</a>
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”.

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, <span style="color: #ff0000;">manages 22%</span> of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&amp;nbsp;</p> <p><a href="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" style="text-align: center;"> icon2</a></p> <p>&amp;nbsp;</p> <p><h3>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</h3></p> <p>&amp;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/>.

4 115. 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/>.)

8 116. 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.

16 117. 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](https://codex.wordpress.org/WordPress_Lessons#Template_Files),  
26 [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
27 and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

28

1 118. 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/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
5 and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

6 119. 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/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
9 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

10 120. 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.

17 121. 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](https://codex.wordpress.org/WordPress_Widgets),  
25 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

26 122. 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           123. 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/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
10 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

11           124. 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](https://codex.wordpress.org/Widgets_API),  
14 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and [https://codex.wordpress.org/Plugin\\_API](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           125. 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](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/...](https://wordpress.org/themes/ultra/))

26           126. 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](https://codex.wordpress.org/Widgets_API),  
7 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

8 127. 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](https://codex.wordpress.org/Widgets_API),  
16 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

17 128. 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/Widgets_API), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and  
23 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

24 129. 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/Widgets_API), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and  
3 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

4 130. 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](http://codex.wordpress.org/Template_Hierarchy);  
8 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
9 [http://codex.wordpress.org/Function\\_Reference/the\\_content](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](http://codex.wordpress.org/Template_Tags/get_the_title); and  
12 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview) .

13 131. 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.

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

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



1           134. 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           135. 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           136. 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           137. 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.

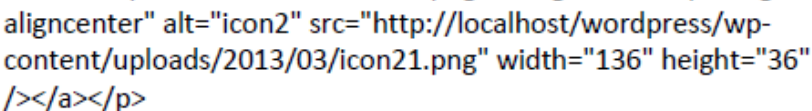
14           138. The Accused Instrumentalities infringe claim 5 of the '287 patent through a  
15 combination of features which collectively practice each limitation of claim 1. By way of example,  
16 the Accused Instrumentalities additionally feature a defined UI object that is an input field for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

18           139. 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           140. 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	<a href="http://localhost/wordpress">http://localhost/wordpress</a>
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, <span style="color: #ff0000;">manages 22%</span> of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&amp;nbsp;</p> <p style="text-align: center;"><a href="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png">http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png</a></p> <p> width="136" height="36"/&gt;</p> <p>&amp;nbsp;</p> <p><b>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</b></p> <p>&amp;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](https://codex.wordpress.org/Widgets_API)).

141. 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           142. 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](https://codex.wordpress.org/Widgets_API).)

5           143. 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           144. 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           145. 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://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/>.)

3 146. 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/WordPress_Widgets) , [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) , and  
12 [https://codex.wordpress.org/Plugin\\_Resources](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/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
18 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

19 147. 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/WordPress_Widgets) , [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) , and  
27 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources).)

28

1           148. 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](https://codex.wordpress.org/WordPress_Widgets),  
13 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

14           149. 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](https://codex.wordpress.org/Widgets_API)) and Plug-  
17 ins (see [https://codex.wordpress.org/Plugin\\_API](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           150. 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](https://codex.wordpress.org/WordPress_Widgets),  
2 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

3 151. 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/WordPress_Widgets) , [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) , and  
14 [https://codex.wordpress.org/Plugin\\_Resources](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](https://codex.wordpress.org/WordPress_Widgets),  
20 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

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

23 153. 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](https://codex.wordpress.org/WordPress_Widgets) ,  
28 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) , and [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources).)

1           154. 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](https://codex.wordpress.org/Widgets_API),  
10 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

11           155. 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](https://codex.wordpress.org/Widgets_API),  
19 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

20           156. 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/Widgets_API), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and  
26 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

27           157. 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/Widgets_API), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and  
6 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

7 158. 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](http://codex.wordpress.org/Template_Hierarchy);  
11 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
12 [http://codex.wordpress.org/Function\\_Reference/the\\_content](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](http://codex.wordpress.org/Template_Tags/get_the_title); and  
15 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview).

16 159. 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.

19 160. 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](https://codex.wordpress.org/Widgets_API);  
25 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources); and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API);  
26 [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets).  
27  
28



1           161. 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           162. 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           163. 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           164. 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           165. 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           166. 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           167. 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           168. 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	<a href="http://localhost/wordpress">http://localhost/wordpress</a>
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, <span style="color: #ff0000;">manages 22%</span> of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&amp;nbsp;</p> <p><a href="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" style="text-align: center;"> icon2" src="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" width="136" height="36"/&gt;</a></p> <p>&amp;nbsp;</p> <p><b>As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.</b></p> <p>&amp;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](https://codex.wordpress.org/Widgets_API)).

1           169. 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           170. 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](https://codex.wordpress.org/Widgets_API).)

7           171. Defendant was made aware of the '287 patent and its infringement thereof at least as  
8 early as the filing of the original Complaint in this action.

9           172. Since the date of the filing of the original Complaint in this action, Defendant's  
10 infringement of the '287 patent has been willful.

11           173. Plaintiff has been harmed by Defendant's infringing activities.

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

13           174. The allegations set forth in the foregoing paragraphs 1 through 173 are incorporated  
14 into this Fourth Claim for Relief.

15           175. The allegations set forth in the foregoing paragraphs 1 through 138 are incorporated  
16 into this Fourth Claim for Relief.

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

20           177. The inventions of the '044 patent resolve technical problems related to generating  
21 content on a display of a device, such as the display of a mobile device. For example, the inventions  
22 feature a computer memory and an authoring tool or Player configured to define a User Interface  
23 ("UI") object for display on the device, where the defined UI object corresponds to a web  
24 component and where each UI object is either: 1) selected by a user or 2) automatically selected by  
25 the system as a preferred UI object corresponding to a symbolic name of the web component.  
26 Additionally, the computer memory and the authoring tool or Player are configured to build an  
27 Application consisting of one or more web page views to provide for the display of at least a portion  
28

1 of one or more of the web pages. These features are exclusively implemented utilizing computer  
2 technology.

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

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

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

16 181. Accordingly, each claim of the '044 patent recites a combination of elements  
17 sufficient to ensure that the claim in practice amounts to significantly more than a patent on an  
18 ineligible concept.

19 182. Plaintiff is the assignee and owner of the right, title and interest in and to the '044  
20 patent, including the right to assert all causes of action arising under the patents and the right to any  
21 remedies for infringement of them.

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

1 more web page views to provide for the display of at least a portion of one or more of the web pages  
2 (the “Accused Instrumentalities”). The Accused Instrumentalities include platforms that enable the  
3 functionality described above and include but are not limited to, for example, WordPress. See, e.g.,  
4 <https://pantheon.io/product/wordpress-hosting?action=>. For example, Defendant’s managed  
5 WordPress hosting services handles the architecture, capacity, setup, and more for its users’  
6 WordPress websites.

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

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

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

16 186. For example, on information and belief, WordPress uses a variety of databases in its  
 17 technology stack including MySQL. Data from the wp\_options table for the website header and the  
 18 data from the wp\_posts table for the "WordPress Info" web page extracted directly from the Bitnami  
 19 WordPress server-side database using MySQL Workbench. The stored data in the wp\_options table  
 20 includes the website's url, the website's title (blogname), the website's tagline (blogdescription), and  
 21 the active template (style sheet).

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

22  
 23  
 24  
 25  
 26  
 27 The stored data in the wp\_posts table for the "WordPress Info" web page includes information  
 28 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”.

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, <span style="color: #ff0000;">manages 22%</span> of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&amp;nbsp;</p> <p style="text-align: center;"><a href="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png">&lt;img class=" wp-image-24 aligncenter" alt="icon2" src="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" width="136" height="36" /&gt;&lt;/a&gt;&lt;/p&gt;</a></p> <p>&amp;nbsp;</p> <p>&lt;h3&gt;As of December 2011, WordPress version 3.0 had been downloaded over 65 million times.&lt;/h3&gt;</p> <p>&amp;nbsp;</p>	WordPress Info	page

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

187. 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/>.)

1           188. The computer memory stores a) symbolic names required for evoking one or more  
2 web components each related to a set of inputs and outputs of a web service obtainable over a  
3 network, where the symbolic names are character strings that do not contain either a persistent  
4 address or pointer to an output value accessible to the web service. The WordPress MySQL  
5 database contains 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. WordPress' WYSIWYG visual effects editor includes elements for defining the  
11 layout for placement of the defined UI objects. Widgets, plug-ins and other elements correspond to  
12 the defined UI objects and are the product of the JSON formatting. (See, for example,  
13 [https://codex.wordpress.org/WordPress\\_Lessons#Template\\_Files](https://codex.wordpress.org/WordPress_Lessons#Template_Files),  
14 [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
15 and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API).)

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

28



1           190. 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. (See,  
3 for example, <https://developer.wordpress.org/rest-api/>,  
4 [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
5 and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

6           191. 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/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
9 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

10           192. 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 computer memory  
12 selected from a group consisting of an input of the web service and an output of the web service.  
13 WordPress's WYSIWYG visual effects editor and widget authoring tools define the presence of a 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           193. 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 (See  
23 [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets) ) steps 1 to 5 under "Displaying Widgets", the  
24 widget UI will automatically displayed in the Web Page Sidebar. (See, for example,  
25 [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
26 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

27  
28

1           194. The authoring tool is configured to access the computer memory to select the  
2 symbolic name corresponding to the web component of the defined UI object based on its JSON  
3 formatting characteristics.

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

14           196. The authoring tool is configured to store information representative of the defined UI  
15 object and related settings in a database. For example, WordPress's computer memory is configured  
16 to store information representative of defined UI objects. (See, for example,  
17 <https://developer.wordpress.org/rest-api/> and [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API).)

18           197. The authoring tool is also configured to retrieve the information representative of the  
19 one or more the UI object settings stored in the database based on the JSON strings. (See, for  
20 example, <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,  
21 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources),  
22 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API), and [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API).)

23           198. The WordPress authoring tool is configured to build an Application consisting of one  
24 or more web page views from at least a portion of the database utilizing at least one Player, where  
25 the Player utilizes information stored in the database to generate for the display of at least a portion  
26 of the one or more web pages. WordPress builds an Application including the symbolic name of the  
27 defined UI object. (See, for example, [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API),  
28 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API),

1 <https://developer.wordpress.org/rest-api/> , <https://developer.wordpress.org/rest-api/reference/>.) It is a  
2 feature of the Accused Instrumentalities that WordPress also contains at least one Player in the form  
3 of a runtime player, such that the Application and the Player are provided to the device and executed  
4 on the device. (See, for example, <https://codex.wordpress.org/CSS> ,  
5 [https://codex.wordpress.org/Using\\_Javascript](https://codex.wordpress.org/Using_Javascript) and <https://developer.wordpress.org/rest-api/> )

6 199. The Accused Instrumentalities feature a system where the Application and Player are  
7 provided to the device and executed on the device.

8 200. When the user of the device provides one or more input values associated with an  
9 input symbolic name to an input of the defined UI object the device provides the user provided one  
10 or more input values and corresponding input symbolic name to the web service. Because the  
11 Accused Instrumentalities incorporate a system that includes WordPress, when a user of the device  
12 provides one or more input values associated with an input symbolic name, using JSON formatting  
13 characteristics, to an input of the defined UI object, the device provides the user provided one or  
14 more input values and corresponding input symbolic name, using JSON formatting characteristics, to  
15 the web service. (See, for example, <https://developer.wordpress.org/rest-api/>,  
16 <https://developer.wordpress.org/rest-api/reference/>, [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API),  
17 [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

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

25 202. The Accused Instrumentalities feature a system where the Player receives the output  
26 symbolic name and corresponding one or more output values and provides instructions for the  
27 display of the device to present an output value in the defined UI object. The runtime player within  
28 WordPress receives the output name, output value, and provides instructions for a display as shown

1 by the fact that the defined UI object are ultimately rendered. (See, for example,  
2 <https://developer.wordpress.org/rest-api/>, <https://developer.wordpress.org/rest-api/reference/>,  
3 [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API), [https://codex.wordpress.org/Plugin\\_Resources](https://codex.wordpress.org/Plugin_Resources), and  
4 [https://codex.wordpress.org/Plugin\\_API](https://codex.wordpress.org/Plugin_API) .)

5 203. The presence of the above referenced features is demonstrated, by way of example,  
6 by reference to publicly available information. Regarding WordPress, *see, e.g.*,  
7 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;  
8 [http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy);  
9 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
10 [http://codex.wordpress.org/Function\\_Reference/the\\_content](http://codex.wordpress.org/Function_Reference/the_content);  
11 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages>;  
12 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Tags/get\\_the\\_title](http://codex.wordpress.org/Template_Tags/get_the_title); and  
13 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview) .

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

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

1           206. Claim 3 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 web  
3 component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.

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

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

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

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

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

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

25           213. The Accused Instrumentalities infringe claim 11 of the '044 patent through a  
26 combination of features which collectively practice each limitation of claim 11. WordPress sends all  
27 files over a network using a variety of databases in its technology stack including MySQL. These  
28 backend capabilities provided the code over a network. By way of example, data from the

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

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

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

ID	post_content	post_title	post_type
23	<p>WordPress is used by over 14.7% of Alexa Internet's "top 1 million" websites and as of August, 2011, believe it or not, <span style="color: #ff0000;">manages 22%</span> of all new websites. WordPress is currently the most popular blogging system in use on the Web.</p> <p>&amp;nbsp;</p> <p style="text-align: center;"><a href="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png">&lt;img class=" wp-image-24 aligncenter" alt="icon2" src="http://localhost/wordpress/wp-content/uploads/2013/03/icon21.png" width="136" height="36" /&gt;&lt;/a&gt;&lt;/p&gt;</a></p> <p>&amp;nbsp;</p> <p style="text-align: center;"><h3&gt;as 2011,="" 3.0="" 65="" been="" december="" downloaded="" h3&gt;<="" had="" million="" of="" over="" p="" times.&lt;="" version="" wordpress=""> <p>&amp;nbsp;</p> </h3&gt;as></p>	WordPress Info	page

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

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

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

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

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

27 218. The computer memory also stores an address of the web service. Because WordPress  
28 contains web services, both as a library of Widgets (see [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API) )

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

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

13 220. Each defined UI object is either: 1) selected by a user of an authoring tool; or 2)  
14 automatically selected by the system as the preferred UI object corresponding to the symbolic name  
15 of the web component selected by the user of the authoring tool. When a Widget is selected in the  
16 WordPress Widget selection list (See [https://codex.wordpress.org/WordPress\\_Widgets](https://codex.wordpress.org/WordPress_Widgets) ) steps 1 to 5  
17 under "Displaying Widgets", the widget UI will automatically displayed in the Web Page Sidebar.  
18 WordPress contains an authoring tool in the form of the WYSIWYG visual effects editor, widgets,  
19 and plug-in authoring tools.

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



1           222. The Accused Instrumentalities also include retrieving the information representative  
2 of the one or more the UI object settings stored in the database based on the JSON formatting  
3 characteristics.

4           223. The Accused Instrumentalities include building an Application consisting of one or  
5 more web page views from at least a portion of the database utilizing the Player, where the Player  
6 utilizes information stored in the database to generate for the display of at least a portion of the one  
7 or more web pages. WordPress builds an Application included in the symbolic name of the defined  
8 UI object.

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

26           225. The presence of the above referenced features is demonstrated, by way of example,  
27 by reference to publicly available information. Regarding WordPress, see, e.g.,  
28 <http://themeforest.net/category/wordpress>; <http://codex.wordpress.org/Templates>;

1 [http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy);  
2 [http://codex.wordpress.org/Function\\_Reference/the\\_title](http://codex.wordpress.org/Function_Reference/the_title);  
3 [http://codex.wordpress.org/Function\\_Reference/the\\_content](http://codex.wordpress.org/Function_Reference/the_content);  
4 <https://www.wpbeginner.com/glossary/database/>; <https://codex.wordpress.org/Pages1>  
5 <http://codex.wordpress.org/Templates>; [http://codex.wordpress.org/Template\\_Tags/get\\_the\\_title](http://codex.wordpress.org/Template_Tags/get_the_title); and  
6 [http://codex.wordpress.org/Query\\_Overview](http://codex.wordpress.org/Query_Overview).

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

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

19 228. Claim 17 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 additionally where the web  
21 component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.  
22

23 229. The Accused Instrumentalities infringe claim 17 of the '044 patent through a  
24 combination of features which collectively practice each limitation of claim 17. By way of example,  
25 the Accused Instrumentalities feature web components additionally including web chat, Reuters RSS  
26 feed, Calendar image, and map image widgets.

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

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

4           232. Claim 19 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 UI object is an input field for a web  
6 service.

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

10           234. Claim 25 of the '044 patent recites a method of displaying content on a display that  
11 includes all the elements of claim 15, additionally where the method includes providing the code over  
12 the network.

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

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

19           237. The Accused Instrumentalities infringe claim 26 of the '044 patent through a  
20 combination of features which collectively practice each limitation of claim 26. By way of example,  
21 the Accused Instrumentalities employ WordPress which includes widgets. See, e.g.,  
22 [https://codex.wordpress.org/Widgets\\_API](https://codex.wordpress.org/Widgets_API).

23           238. Defendant was made aware of the '044 patent and its infringement thereof at least as  
24 early as the filing of the original Complaint in this action.

25           239. Since the date of the filing of the original Complaint in this action, Defendant's  
26 infringement of the '044 patent has been willful.

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



1  
2 Dated: October 15, 2018

LOCAL COUNSEL

3 By: /s/ Jeffrey Francis Craft  
4 Jeffrey Francis Craft (SBN 147186)  
5 DEVLIN LAW FIRM  
6 2069 Cold Canyon Road  
7 Calabasas, CA 91302

8 Robert Kiddie (*pro hac vice* to be filed)  
9 Texas Bar No. 24060092  
10 DEVLIN LAW FIRM LLC  
11 1306 N. Broom Street, 1<sup>st</sup> Floor  
12 Wilmington, DE 19806  
13 Phone: (302) 449-9010  
14 rkiddie@devlinlawfirm.com

*Attorneys for Plaintiff Express Mobile, Inc.*

15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  

**CERTIFICATE OF SERVICE**

This is to certify that a true and correct copy of this document has been served on all parties through counsel of record on this October 15, 2018 via the Court’s CM/ECF system.

By: /s/ Jeffrey Francis Craft  
Jeffrey Francis Craft SBN 147186  
jcraft@devlinlawfirm.com  
DEVLIN LAW FIRM LLC  
1306 N. Broom St., 1st Floor  
Wilmington, DE 19806  
Tel. (302) 449-9010  
Fax. (302) 353-4251  
Attorney for Plaintiff