

IN THE UNITED STATES DISTRICT COURT  
FOR THE SOUTHERN DISTRICT OF WEST VIRGINIA

CHARLESTON DIVISION

CSS, INC.,

Plaintiff,

v.

CIVIL ACTION NO. 2:16-cv-01762

CHRISTOPHER HERRINGTON, et al.,

Defendants.

**MEMORANDUM OPINION AND ORDER**

Pending before the court is the Defendants' Motion for Summary Judgment [ECF No. 123]. For the following reasons, the Defendants' Motion is **GRANTED in part** and **DENIED in part**.

**I. Background**

On February 23, 2016, CSS, Inc. ("CSS") filed a Complaint [ECF No. 1] against Christopher Herrington, Gene Yoho, and Compiled Technologies, LLC (collectively, "Defendants"). On September 16, 2016, CSS filed its First Amended Complaint [ECF No. 59] against the Defendants, alleging copyright infringement, breach of contract, violation of the duty of loyalty, tortious interference with business relationships, unjust enrichment, and misappropriation of trade secrets.

CSS is a company that provides software and related support services to county clerks' offices in West Virginia. *See* Hr'g Tr., Jan. 25, 2017, Auburn Direct, 28:1–9. For over twenty years, CSS has provided software solutions that include a land

records indexing package, an estate management package, utility billing, and sheriff's tax collection applications, among others. *See id.* at 28:1–4.

Christopher Herrington began his employment with CSS in October 1991. *See Hr'g Tr.*, Feb. 23, 2017, C. Herrington Direct, 157:8–12, 190:14–15. Herrington worked in the computer software, applications, and programming business related to county government information and document management systems. *See id.* at 157:8–12. For the majority of his employment with CSS, Herrington worked as a programmer/developer. *See Hr'g Tr.*, Jan. 25, 2017, Auburn Direct, 14:14–23, 16:22–17:5. Herrington's work consisted of the development of CSS's software, bug fixes, modifications, and providing support services for CSS's software packages. *Id.* at 14:14–23, 15:13–23, 16:22–17:5. Specifically, his duties included programming and support responsibilities for both the land indexing software and the estate management software. *Id.* at 16:2–9.

Christopher Herrington resigned as an employee of CSS on August 23, 2014. *See Hr'g Tr.*, Feb. 23, 2017, C. Herrington Direct, 190:14–15. On September 6, 2014, Herrington returned to CSS. *See id.* at 203:22–24. On September 12, 2014, CSS required Christopher Herrington to sign a Confidentiality Agreement. *See Hr'g Tr.*, Jan. 25, 2017, Auburn Direct, 44:14–45:2. Prior to September 12, 2014, CSS had never required Herrington to sign a confidentiality agreement. *See Hr'g Tr.*, Feb. 23, 2017, McCasker Redirect, 97:6–17. CSS employed Christopher Herrington until June 5, 2015. *See Hr'g Tr.*, Feb. 23, 2017, C. Herrington Direct, 203:22–24.

Gene Yoho formed Compiled Technologies, LLC (“CT”), on August 12, 2015. *See Hr’g Tr.*, Feb. 24, 2017, C. Herrington Direct, 43:19–44:5. Gene Yoho and Christopher Herrington operate CT. *See id.* CT provides licensing for custom computer software applications, including one program for land records recording and indexing and one program for estate management. *See id.* at 44:3–5. CT also provides support services to county clerks’ offices related to these applications. *See Hr’g Tr.*, Feb. 24, 2017, C. Herrington Cross, 242:9–11. CT competes with CSS for contracts with West Virginia county clerks. *See Hr’g Tr.*, Feb. 24, 2017, C. Herrington Direct, 43:19–44:5.

In its First Amended Complaint, CSS alleges that the Defendants infringed various copyrights relating to its computer software code for the land indexing and estate management applications. First Am. Compl. ¶¶ 32–39. Specifically, CSS alleges that CT’s software programs, which were developed by Christopher Herrington, are substantially similar to CSS’s copyrighted programs. CSS also alleges that Christopher Herrington breached the Confidentiality Agreement by using confidential information to create, market, and sell software and support services to CSS’s customers. *Id.* at ¶ 43. CSS alleges that Herrington violated the duty of loyalty by creating competing products, and soliciting CSS’s customers to use those competing products, while still employed by CSS. *Id.* at ¶¶ 47–50. In addition, CSS alleges that the Defendants tortiously interfered with CSS’s business relationships with actual and potential customers, that the Defendants misappropriated CSS’s trade secrets, and that the Defendants were unjustly enriched

by their unlawful use and disclosure of confidential information. *Id.* at ¶¶ 52–65. Finally, CSS seeks a declaratory judgment that, under the “works for hire” doctrine, CSS is the author of any computer programs developed by Christopher Herrington during his employment with CSS. *Id.* at ¶ 75.

On November 23, 2016, CSS filed Plaintiff’s Motion for Preliminary Injunction [ECF No. 65], wherein it asserted that it was likely to succeed on the merits of its copyright infringement, misappropriation of trade secrets, and breach of contract claims. The court conducted a hearing on the Plaintiff’s Motion on January 25, February 23, and February 24, 2017. On August 1, 2017, the court issued a Memorandum Opinion and Order [ECF No. 209] denying the Plaintiff’s Motion. On August 4, 2017, the court issued an Amended Memorandum Opinion and Order (“Prelim. Inj. Op.”) [ECF No. 213] that corrected clerical errors in the earlier Opinion. In denying the Plaintiff’s Motion, the court found that CSS was not likely to succeed on its claims for copyright infringement, misappropriation of trade secrets, or breach of contract.

On April 27, 2017, the Defendants filed a Motion for Summary Judgment [ECF No. 123] and a Memorandum in Support of the motion [ECF No. 124]. On May 11, 2017, CSS filed a Response to Defendants’ Motion for Summary Judgment [ECF No. 130]. On May 18, 2017, the Defendants filed their Reply [ECF No. 134]. Thus, the Defendants’ Motion is ripe for adjudication.

## II. Legal Standard

To obtain summary judgment, the moving party must show that there is no genuine dispute as to any material fact and that the moving party is entitled to judgment as a matter of law. Fed. R. Civ. P. 56(a). In considering a motion for summary judgment, the court will not “weigh the evidence and determine the truth of the matter.” *Anderson v. Liberty Lobby, Inc.*, 477 U.S. 242, 249 (1986). Instead, the court will draw any permissible inference from the underlying facts in the light most favorable to the nonmoving party. *Matsushita Elec. Indus. Co. v. Zenith Radio Corp.*, 475 U.S. 574, 587–88 (1986).

Although the court will view all underlying facts and inferences in the light most favorable to the nonmoving party, the nonmoving party nonetheless must offer some “concrete evidence from which a reasonable juror could return a verdict” in his or her favor. *Anderson*, 477 U.S. at 256. Summary judgment is appropriate when the nonmoving party has the burden of proof on an essential element of his or her case and does not make, after adequate time for discovery, a showing sufficient to establish that element. *Celotex Corp. v. Catrett*, 477 U.S. 317, 322–23 (1986). The nonmoving party must satisfy this burden of proof by offering more than a mere “scintilla of evidence” in support of his or her position. *Anderson*, 477 U.S. at 252. Likewise, conclusory allegations or unsupported speculation, without more, are insufficient to preclude the granting of a summary judgment motion. *See Dash v. Mayweather*, 731 F.3d 303, 311 (4th Cir. 2013); *Stone v. Liberty Mut. Ins. Co.*, 105 F.3d 188, 191 (4th Cir. 1997).

### III. Discussion

#### A. Copyright Infringement

“Copyright protection subsists . . . in original works of authorship fixed in any tangible medium of expression . . . from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.” 17 U.S.C. § 102(a). Computer programs can be protected as “literary works.” *See* H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5667 (“The term ‘literary works’ does not connote any criterion of literary merit or qualitative value: it includes . . . computer data bases, and computer programs to the extent that they incorporate authorship in the programmer’s expression of original ideas, as distinguished from the ideas themselves.”). Indeed, “copyright protection can extend to both literal and non-literal elements of a computer program.” *Oracle Am., Inc. v. Google Inc.*, 750 F.3d 1339, 1355 (Fed. Cir. 2014). “The literal elements of a computer program are the source code and object code.” *Id.* “The non-literal components of a computer program include, among other things, the program’s sequence, structure, and organization, as well as the program’s user interface.” *Id.* at 1355–56.

However, “[t]he mere fact that a work is copyrighted does not mean that every element of the work may be protected.” *Feist Publ’ns, Inc. v. Rural Tel. Serv. Co.*, 499 U.S. 340, 348 (1991). “In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described,

explained, illustrated, or embodied in such work.” 17 U.S.C. § 102(b). In the case of computer programs, this limitation “is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law.” H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5670.

“Originality remains the *sine qua non* of copyright; accordingly, copyright protection may extend only to those components of a work that are original to the author.” *Feist*, 499 U.S. at 348. “[H]owever, the originality requirement is not particularly stringent.” *Id.* at 358. “Original, as the term is used in copyright, means only that the work was independently created by the author (as opposed to copied from other works), and that it possesses at least some minimal degree of creativity.” *Id.* at 345.

### **1. CSS’s and CT’s Computer Programs**

CSS’s claims of copyright infringement pertain to three of its computer programs: CSS-Index, CSS-Estate, and CSS-Search. Each of these programs corresponds, in terms of function, to one of CT’s programs: CT-Index, CT-Estate, and CT-Search, respectively. The Index programs create and maintain indexes for recorded documents. Mot. by Defs. for Leave to File Under Seal Ex. 1, at 3 [ECF No. 70-1] (“McCasker Expert Report”). The Estate programs process information related to decedents. *Id.* The Search programs allow the public to search for indexed documents online. *Id.*

Both CSS's and CT's programs use a client/server architecture. Client/server describes "a program relationship in which one program (the client) requests a service or resource from another program (the server)." Mot. by Defs. for Leave to File Under Seal Ex. 2 ("Zeidman Expert Report"), Ex. J at 1 [ECF No. 70-2]. The "client side" is also called the "front end," while the "server side" is also called the "back end."

The parties' programs use three programming languages: COBOL, Microsoft Visual Basic Version 6 ("VB6"), and C#. COBOL is an early programming language, introduced in 1959, that was widely used in the 1970s and 1980s. *See* Zeidman Expert Report 3; McCasker Expert Report 25. COBOL is "exclusively owned, developed and maintained by Micro Focus," and thus is not the intellectual property of any of the parties in this case. Defs.' Mem. Opp. Pl.'s Mot. Prelim. Inj. Ex. 9, at 2 [ECF No. 68-9] ("Letter from Courtney Wood, Associate General Counsel for Micro Focus"). The server sides of both CSS's and CT's programs are written in COBOL. Specifically, CSS uses RM/COBOL version 11, and CT uses RM/COBOL version 12. CSS's client-side applications are written in VB6, while CT's client-side applications are written in C#.

CSS-Index contains the following architecture components:

- CSS-Index Client Application: presents the screens and other user interface components; accepts user inputs; displays responses back to the user.
- CSS-Search Client Application: launches when a user clicks "Inquiry" from within the CSS-Index Client Application.



- CSS Network Communication Interface (VanGUI): bridges communication between the client application and the back-end COBOL programs; runs over Transmission Control Protocol (“TCP”) sockets; built using VanGUI interface.
- CSS-Index Application Server: reads, writes, updates, and deletes records in the permanent COBOL data files; writes records to temporary COBOL data files.
- CSS-Index Persistent Data Repository: comprises the permanent COBOL data files. When a user enters information in the CSS-Index Client Application and clicks “save,” the information is sent to a COBOL program that manipulates the information and saves it into one or more permanent data files.
- CSS Report Engine (Crystal Reports): uses report definition files to execute reports, which in turn access data using an Open Database Connectivity (“ODBC”) interface.
- CSS Data Bridge (Relativity Server): provides an application with access to COBOL data files via an ODBC interface.
- CSS-Index Reporting Data Repository: temporary data files that contain the data that will be used in a report requested by the user.

McCasker Expert Report 7–9.

CT-Index contains the following architecture components: CT-Index Client Application; CT-Search Client Application; CT Network Communication Interface (Louis); CT-Index Application Server; CT-Index Persistent Data Repository; CT Report Engine (Crystal Reports); CT Data Bridge (Relativity Server); CT-Index

Reporting Data Repository. *Id.* at 9–11. The description of each of these components is essentially the same as that of the corresponding component of CSS-Index described above. *See id.* One notable difference is that the CSS Network Communication Interface uses the communication middleware VanGUI, while the CT Network Communication Interface uses Louis.

VanGUI is a communication middleware that allows back-end applications written in COBOL to communicate with graphical user interface (“GUI”) front-end applications written in programming languages that support VB6. Letter from Courtney Wood, Associate General Counsel for Micro Focus 6–7. VanGUI is a third-party component that is not the intellectual property of any of the parties in this case. Louis is an interface builder that works in conjunction with COBOL platforms and GUIs. Louis is “exclusively owned, developed and maintained by Micro Focus,” and therefore is not the intellectual property of any of the parties in this case. *Id.* at 2.

The other third-party components used by both of these programs are Relativity and Crystal Reports. Relativity is a data access middleware product that improves access to data stored in COBOL applications. *Id.* at 4–5. Specifically, Relativity is an ODBC-accessible relational database management system. *Id.* It is a COBOL-specific database engine/driver that is “exclusively owned, developed and maintained by Micro Focus.” *Id.* at 2. Crystal Reports is a business intelligence application that helps generate reports, and is not the intellectual property of any of the parties in this case. Zeidman Expert Report, Ex. H at 2.

The CSS-Search program can either be launched from within the CSS-Index program, or it can run independently. McCasker Expert Report 13. The same is true of the CT-Search program. *Id.* The architecture components of these two programs have already been described within the description of the CSS-Index and CT-Index programs. *See id.* The architecture components of the CSS-Estate and CT-Estate programs are very similar to those of the CSS-Index and CT-Index programs, as described above, so I will not repeat them here. *Id.* at 14–15.

Two particular features of these programs require further explanation. First, both companies' programs contain a name search algorithm that allows a user to search for records stored on the server using an individual's name. *Id.* at 23. Both programs store an individual's name in the permanent COBOL data files in two different ways: exactly as entered by the user (e.g., first, middle, and last names), and with all spaces and punctuation removed ("letters-only"). *Id.* When a user initiates a search using the individual's first, middle, and/or last name, the server converts the name into the letters-only version to conduct the search through the data files. *Id.*

Second, both companies' programs contain a single entry point to perform both login validation and password change. *Id.* at 24. In both cases, the client application requests the following information from the user: username, password, new password, confirm new password. *Id.* at 23–24. The server then validates the login by comparing the username and password against records in a COBOL data file. *Id.* at 24. If the login is validated, the program looks to see if a new password was entered. *Id.* If so, the program compares the "new password" entry with the "confirm new

password” entry. *Id.* If the entries match, the server updates the password in the COBOL data files with the new password and then exits. *Id.*

## 2. CSS’s Copyright Ownership and the Defendants’ Access

“To establish a claim for copyright infringement, a plaintiff must prove that it owned a valid copyright and that the defendant copied the original elements of that copyright.” *Humphreys & Partners Architects, L.P. v. Lessard Design, Inc.*, 790 F.3d 532, 537 (4th Cir. 2015) (quoting *Lyons P’ship, L.P. v. Morris Costumes, Inc.*, 243 F.3d 789, 801 (4th Cir. 2001)). A certificate of copyright registration is prima facie evidence of the validity of the copyright. 17 U.S.C. § 410(c).

CSS is the registered owner of three copyrights relating to its computer programs: Document Indexing and Imaging for Counties 2015 (TX 8-230-758) (i.e., CSS-Index), Estate Management 2015 (TX 8-230-809) (i.e., CSS-Estate), and Web Inquiry 2013 (TX 8-230-816) (i.e., CSS-Search). The Defendants do not dispute the validity of these copyrights in their Motion for Summary Judgment or Memorandum in Support. Therefore, I **FIND** that CSS owns valid copyrights in its computer programs, and I turn to the issue of copying.

CSS alleges that the Defendants committed copyright infringement by copying the non-literal elements of its computer software.<sup>1</sup> To establish copying through circumstantial evidence, CSS must prove that the Defendants “had access to the work and that the supposed copy is substantially similar to [CSS’s] original work.” *Humphreys*, 790 F.3d at 537 (quoting *Bouchat v. Balt. Ravens, Inc.*, 241 F.3d 350,

---

<sup>1</sup> CSS concedes that the Defendants did not literally copy CSS’s source code.

353–54 (4th Cir. 2001)). The Defendants do not dispute that they had access to CSS’s software code, and the evidence of such access is clear. Christopher Herrington worked for CSS as a programmer/developer for almost twenty-four years. He was responsible for developing computer programs, programming software solutions, and servicing the software for CSS’s clients. In the course of his employment with CSS, Christopher Herrington worked directly with CSS’s copyrighted computer programs. Accordingly, I **FIND** that the Defendants had access to CSS’s copyrighted works. I now must determine whether the Defendants’ computer programs are substantially similar to CSS’s original works.

### **3. Substantial Similarity and the AFC Test**

CSS argues that CT’s software programs copy the structure, architecture, and transactional flow of CSS’s software programs. In his report, CSS’s expert, Andrew McCasker, found that the two Index programs use the same architecture, the two Estate programs use the same architecture, and the two Search programs use the same architecture. *Id.* at 26. In addition, CSS notes that both companies’ programs contain the following elements: use of the COBOL programming language on the back end, COBOL data files, temporary data files, Crystal Reports, Relativity, “the same or nearly identical network interface used to communicate with the back end,” “the same grouping for its business logic and reporting logic,” “the same unique back-end structure to [their] log-in and password change functions,” “and a unique name stripping search algorithm.” Pl.’s Resp. Defs.’ Mot. Summ. J. 16.

I will begin by noting “that the application of copyright law in the computer context is often a difficult task.” *Oracle Am.*, 750 F.3d at 1354. The Fourth Circuit has not yet established a test for determining substantial similarity of the non-literal elements of computer programs.<sup>2</sup> Many jurisdictions endorse the Second Circuit’s abstraction-filtration-comparison test (“AFC Test”) to address this issue. *See Comput. Assocs. Int’l, Inc. v. Altai, Inc.*, 982 F.2d 693, 706–11 (2d Cir. 1992) (outlining the AFC test). The Fifth, Ninth, Tenth, and Eleventh Circuits apply the AFC Test. *See Gen. Universal Sys., Inc. v. Lee*, 379 F.3d 131, 142 (5th Cir. 2004); *MiTek Holdings, Inc. v. Arce Eng’g Co.*, 89 F.3d 1548, 1559 (11th Cir. 1996); *Gates Rubber Co. v. Bandon Chem. Indus., Ltd.*, 9 F.3d 823, 833 (10th Cir. 1993); *Sega Enters. Ltd. V. Accolade, Inc.*, 977 F.2d 1510, 1525 (9th Cir. 1992). The Sixth Circuit also uses a similar test. *See Lexmark Int’l, Inc. v. Static Control Components, Inc.*, 387 F.3d 522, 534 (6th Cir. 2004). The First Circuit, in a case of first impression, viewed the AFC Test favorably, but ultimately did not rely on it because the case involved literal (rather than non-literal) infringement of computer software. *See Lotus Dev. Corp. v. Borland Int’l Inc.*, 49 F.3d 807, 814–15 (1st Cir. 1995).

The Second Circuit designed the AFC Test to address the complexities of determining substantial similarity that “result[] from the hybrid nature of a computer program, which, while it is literary expression, is also a highly functional,

---

<sup>2</sup> In *Comprehensive Techs. Int’l, Inc. v. Software Artisans, Inc.*, 3 F.3d 730, 734–35 (4th Cir. 1993), the appellant argued that the district court erred in failing to apply the abstraction-filtration-comparison test. However, the Fourth Circuit affirmed the district court’s decision without addressing this issue. *Id.* at 735.

utilitarian component in the larger process of computing.” *Comput. Assocs. Int’l*, 982

F.2d at 712. Essentially, the test is applied as follows:

In ascertaining substantial similarity under this approach, a court would first break down the allegedly infringed program into its constituent structural parts. Then, by examining each of these parts for such things as incorporated ideas, expression that is necessarily incidental to those ideas, and elements that are taken from the public domain, a court would then be able to sift out all non-protectable material. Left with a kernel, or possible kernels, of creative expression after following this process of elimination, the court’s last step would be to compare this material with the structure of an allegedly infringing program. The result of this comparison will determine whether the protectable elements of the programs at issue are substantially similar so as to warrant a finding of infringement.

*Id.* at 706.<sup>3</sup>

As the Second Circuit recognized, “[t]his approach breaks no new ground; rather, it draws on such familiar copyright doctrines as merger, *scenes a faire*, and public domain.” *Id.* Indeed, the AFC Test “has the major advantage of being entirely consistent with the infringement calculus used for infringement issues concerning other types of works. In effect, it is an adaptation of the traditional infringement test to the medium of computer programs.” Howard B. Abrams, *Copying of Protected*

---

<sup>3</sup>

Importantly, this full analysis only applies where a copyright owner alleges infringement of the non-literal aspects of its work. Where “admitted literal copying of a discrete, easily-conceptualized portion of a work” is at issue . . . a court “need not perform a complete abstraction-filtration-comparison analysis” and may focus the protectability analysis on the filtration stage, with attendant reference to standard copyright principles.

*Oracle Am.*, 750 F.3d at 1357 n.4 (quoting *Mitel, Inc. v. Iqtel, Inc.*, 124 F.3d 1366, 1372–73 (10th Cir. 1997)).

*Expression—Technology, Functionality and the Ordinary Observer Test—*  
*Commentary*, 2 The Law of Copyright § 14:33 (2016). Accordingly, I will apply the AFC Test to determine whether CT’s software programs are substantially similar to the protectable elements of CSS’s software programs.

**i. Step One: Abstraction**

The first step in determining whether the non-literal elements of the computer programs are substantially similar is abstraction.

Initially, in a manner that resembles reverse engineering on a theoretical plane, a court should dissect the allegedly copied program’s structure and isolate each level of abstraction contained within it. This process begins with the code and ends with an articulation of the program’s ultimate function. Along the way, it is necessary essentially to retrace and map each of the designer’s steps—in the opposite order in which they were taken during the program’s creation.

*Comput. Assocs. Int’l*, 982 F.2d at 707.

At the lowest level of abstraction, a computer program may be thought of in its entirety as a set of individual instructions organized into a hierarchy of modules. At a higher level of abstraction, the instructions in the lowest-level modules may be replaced conceptually by the functions of those modules. At progressively higher levels of abstraction, the functions of higher-level modules conceptually replace the implementations of those modules in terms of lower-level modules and instructions, until finally, one is left with nothing but the ultimate function of the program. . . . A program has structure at every level of abstraction at which it is viewed. At low levels of abstraction, a program’s structure may be quite complex; at the highest level it is trivial.



*Id.* (ellipsis in original) (quoting Steven R. Englund, Note, *Idea, Process, or Protected Expression?: Determining the Scope of Copyright Protection of the Structure of Computer Programs*, 88 Mich. L. Rev. 866, 897–98 (1990)).

At the highest level of abstraction, each of the allegedly copied programs and their corresponding alleged copies can be described in terms of their functions: The Index programs create and maintain indexes for recorded documents; the Estate programs process information related to a decedent; and the Search programs allow the public to search for indexed documents online. McCasker Expert Report 3. At the next lower level of abstraction, each of the programs can be described in terms of their architecture: Each program uses client/server architecture, with the back end written in COBOL and the front end written in either VB6 (CSS) or C# (CT).

At the next lower level of abstraction, both the front end and back end of each program can be distilled into its component parts, which I have described in detail above. Using the CSS-Index program as an example, the front end is comprised of the following components: CSS-Index Client Application, CSS-Search Client Application, CSS Network Communication Interface (VanGUI), CSS Report Engine (Crystal Reports), and the ODBC Interface. The back end of CSS-Index is comprised of the following components: CSS Network Communication Interface (VanGUI), CSS-Index Application Server, CSS-Index Persistent Data Repository, CSS Data Bridge (Relativity), and CSS-Index Reporting Data Repository. The front end communicates with the back end using the CSS Network Communication Interface (VanGUI) which runs over TCP sockets. The front and back ends of the CT-Index program are

comprised of essentially the same components as the front and back ends of the CSS-Index program, with the exception that CT uses Louis as its Network Communication Interface, rather than VanGUI.

Each of these component parts can be broken down into lower levels of abstraction, based on the various functions they perform. At the lowest level of abstraction, the court would be left with a set of discrete instructions each comprised of lines of source code. Because CSS's allegations of copyright infringement focus on structure, architecture, and transactional flow, I will discuss these lower levels of abstraction only to the extent necessary in the subsections that follow.

#### ii. Step Two: Filtration

The second step in determining substantial similarity is filtration, which “serves ‘the purpose of defining the scope of plaintiff’s copyright.’” *Comput. Assocs. Int’l*, 982 F.2d at 707 (quoting *Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465, 1475 (9th Cir. 1992)).

This process entails examining the structural components at each level of abstraction to determine whether their particular inclusion at that level was “idea” or was dictated by considerations of efficiency, so as to be necessarily incidental to that idea; required by factors external to the program itself; or taken from the public domain and hence is nonprotectable expression.

*Id.* The filtration process draws on familiar considerations of traditional copyright analysis.

“It is a fundamental principle of copyright law that a copyright does not protect an idea, but only the expression of the idea.” *Id.* at 703; *see also* 17 U.S.C. § 102(b). Furthermore, “those aspects of a work, which ‘must necessarily be used as incident

to' the idea, system or process that the work describes, are also not copyrightable.” *Comput. Assocs. Int'l*, 982 F.2d at 704 (quoting *Baker v. Selden*, 101 U.S. 99, 104 (1879)). Applying this principle to software, “those elements of a computer program that are necessarily incidental to its function are similarly unprotectable.” *Id.* at 705.

“The merger doctrine functions as an exception to the idea/expression dichotomy. It provides that, when there are a limited number of ways to express an idea, the idea is said to ‘merge’ with its expression, and the expression becomes unprotected.” *Oracle Am.*, 750 F.3d at 1359 (citing *Comput. Assocs. Int'l*, 982 F.2d at 707–08).

In the context of computer program design, the concept of efficiency is akin to deriving the most concise logical proof or formulating the most succinct mathematical computation. Thus, the more efficient a set of modules are, the more closely they approximate the idea or process embodied in that particular aspect of the program’s structure.

While, hypothetically, there might be a myriad of ways in which a programmer may effectuate certain functions within a program,—i.e., express the idea embodied in a given subroutine—efficiency concerns may so narrow the practical range of choice as to make only one or two forms of expression workable options.

*Comput. Assocs. Int'l*, 982 F.2d at 708. Thus,

in order to determine whether the merger doctrine precludes copyright protection to an aspect of a program’s structure that is so oriented, a court must inquire “whether the use of *this particular set* of modules is necessary efficiently to implement that part of the program’s process” being implemented. If the answer is yes, then the expression represented by the programmer’s choice of a specific module or group of modules has merged with their underlying idea and is unprotected.

*Id.* (citations omitted) (citing Englund, *supra* at 902–03); *see also Oracle Am.*, 750 F.3d at 1360 (“[T]he ‘unique arrangement of computer program expression . . . does not merge with the process so long as alternate expressions are available.’” (ellipsis in original) (quoting *Atari Games Corp. v. Nintendo of Am. Inc.*, 975 F.2d 832, 840 (Fed. Cir. 1992))).

Another familiar principle of traditional copyright law is the *scenes a faire* doctrine. This doctrine “provides that ‘expressive elements of a work of authorship are not entitled to protection against infringement if they are standard, stock, or common to a topic, or if they necessarily follow from a common theme or setting.’” *Oracle Am.*, 750 F.3d at 1363 (quoting *Mitel, Inc. v. Iqtel, Inc.*, 124 F.3d 1366, 1374 (10th Cir. 1997)). “In the computer context, the scene[s] a faire doctrine denies protection to program elements that are dictated by external factors such as the mechanical specifications of the computer on which a particular program is intended to run or widely accepted programming practices within the computer industry.” *Id.* (internal quotation marks omitted) (quoting *Softel, Inc. v. Dragon Med. & Sci. Commc’ns, Inc.*, 118 F.3d 955, 963 (2d Cir. 1997)).

Finally, it is well established in copyright law that material found in the public domain “is free for the taking and cannot be appropriated by a single author even though it is included in a copyrighted work.” *Comput. Assocs. Int’l*, 982 F.2d at 710.

Thus, at the filtration step of the AFC Test, I must examine CSS’s copyrighted programs and filter out ideas, expression dictated by efficiency, elements dictated by external factors, and material found in the public domain. In this way, copyright

protection for the non-literal elements of a computer program can be considered analogous to that for compilations. As explained by the Supreme Court in *Feist*, “This inevitably means that the copyright in a factual compilation is thin. Notwithstanding a valid copyright, a subsequent compiler remains free to use the facts contained in another’s publication to aid in preparing a competing work, so long as the work does not feature the same selection and arrangement.” *Feist*, 499 U.S. at 349. Similarly, in the case of computer programs, a subsequent programmer remains free to use the ideas and industry-standard practices contained in a copyrighted software program to prepare a competing program, as long as the competing program does not copy the copyrighted work’s original expression of those ideas or standard practices. *See Oracle Am.*, 750 F.3d at 1367 (“Oracle does not—and concedes that it cannot—claim copyright in the idea of organizing functions of a computer program or in the ‘package-class-method’ organizational structure in the abstract. Instead, Oracle claims copyright protection only in its *particular* way of naming and organizing each of the 37 Java API packages.”).

Turning to CSS’s software programs, at the highest level of abstraction, I must filter out the function that each program performs, because these are the “ideas” of the programs, not their expression. At the next level of abstraction, I must filter out the client/server architecture used by each of the programs, because this is an industry-standard practice, and thus is not original. *See Zeidman Expert Report*, Ex. J at 2 (“Computer transactions in which the server fulfills a request made by a client

are very common and the client/server model has become one of the central ideas of network computing.”).

At the next level of abstraction, I must filter out all of the third-party components in CSS’s programs: the COBOL programming language, VanGUI, Relativity, and Crystal Reports. As elements in the public domain, these components are “free for the taking and cannot be appropriated by a single author even though [they are] included in a copyrighted work.” *Comput. Assocs. Int’l*, 982 F.2d at 710.

While CSS concedes that it cannot assert a claim of copyright infringement with respect to these third-party components, it argues that “[t]he selection and arrangement of the components, however, can be a portion of the copyright claim. . . . In creating the structure and flow of its software program[s], CSS made numerous decisions which demonstrate creativity.” Pl.’s Resp. Defs.’ Mot. Summ. J. 15. CSS is correct, of course, that it may receive copyright protection for an original selection and arrangement of components that, individually, are unprotectable. However, I disagree with CSS that its particular selection and arrangement of third-party components contains “the modicum of creativity necessary to transform mere selection into copyrightable expression.” *Feist*, 499 U.S. at 362.

Mr. McCasker testified that “the choice of the [programming] language dictated . . . the choice of components in between[, which] in turn drive design decisions . . . where the design ends up being the same.” Hr’g Tr., Jan. 25, 2017, McCasker Direct, 125:2–5. Similarly, the Defendants’ expert, Robert Zeidman, notes in his report that “[t]hese third-party programs require connections to the rest of the

program in certain ways that limit the number of ways that such a program can be structured.” Zeidman Expert Report 14. Therefore, just like arranging names alphabetically in a white pages directory (as was the case in *Feist*), I **FIND** that CSS’s selection and arrangement of third-party components in its software programs “is not only unoriginal, it is practically inevitable.” *Feist*, 499 U.S. at 363. Accordingly, neither the third-party components themselves, nor their particular arrangement within CSS’s programs, are entitled to copyright protection.

Finally, I will address the name search algorithm and login validation/password change features that CSS’s and CT’s programs share. In the name search algorithm, the use of a “letters-only” style search is an idea, not copyrightable expression. Moreover, this type of search algorithm is unoriginal. *See* Zeidman Expert Report 32 (“This method of searching for names by letters only is well known in the field.”). Therefore, the method of searching data files by removing spaces and punctuation is not copyrightable.

Similarly, in the case of the login validation/password change algorithm, the decision to combine the login validation and the password change features into a single entry point or “action” is an idea, not copyrightable expression. And again, this combination of features is unoriginal. Mr. McCasker testified at the hearing that “Windows XP actually has the same kind of user interface.” Hr’g Tr., Jan. 25, 2017, McCasker Direct, 138:3–4. Mr. Zeidman noted in his report that “[t]his is evidence that this technique is well-known in the industry.” Zeidman Expert Report 33.

Therefore, the combination of the login validation and password change features into a single entry point is not copyrightable.

### iii. Step Three: Comparison

The final step in determining substantial similarity is comparison.

Once a court has sifted out all elements of the allegedly infringed program which are “ideas” or are dictated by efficiency or external factors, or taken from the public domain, there may remain a core of protectable expression. . . . At this point, the court’s substantial similarity inquiry focuses on whether the defendant copied any aspect of this protected expression, as well as an assessment of the copied portion’s relative importance with respect to the plaintiff’s overall program.

*Comput. Assocs.*, 982 F.2d at 710.

Now that I have “sifted out” all of the unprotected elements of CSS’s programs, I am left with the “core of protected expression,” and I must compare this protected expression to CT’s allegedly infringing programs to determine whether they are substantially similar. However, CSS has not produced any evidence of substantial similarity beyond the unprotected ideas that I have already filtered out. Nor does Mr. McCasker’s report identify any similarities between CSS’s and CT’s programs that could qualify as protected expression. Therefore, I **FIND** that, with respect to the *protected expression* contained within CSS’s three software programs, CT’s programs are not substantially similar.

### 4. Conclusion

“While the question of substantial similarity ‘typically should be left to the factfinder, summary judgment may be appropriate if the court can conclude, after viewing the evidence and drawing inferences in a manner most favorable to the



nonmoving party, that no reasonable juror could find substantial similarity.” *Nola Spice Designs, LLC v. Haydel Enters., Inc.*, 783 F.3d 527, 550 (5th Cir. 2015) (quoting *Peel & Co. v. Rug Mkt.*, 238 F.3d 391, 395 (5th Cir. 2001)). Given the complete lack of evidence of any similarity regarding the protected elements of CSS’s and CT’s computer programs, I **FIND** that no reasonable jury could conclude that CSS’s and CT’s programs are substantially similar in protectable expression. I also **FIND** that there is no genuine dispute of any material fact and that the Defendants are entitled to judgment as a matter of law on the copyright infringement claim. Accordingly, the Defendants’ Motion for Summary Judgment is **GRANTED** with respect to **Count I** (Copyright Infringement) of the First Amended Complaint.

#### **B. Misappropriation of Trade Secrets**

CSS alleges that the Defendants misappropriated certain trade secrets in creating and selling their competing software. In order to determine whether misappropriation occurred, I must first determine whether there was a trade secret to be protected and then, if one exists, determine whether it was misappropriated. *McGough v. Nalco Co.*, 496 F. Supp. 2d 729, 739 (N.D. W. Va. 2007) (citing *Servo Corp. v. Gen. Elec. Co.*, 393 F.2d 551, 555 (4th Cir. 1968)).

The West Virginia Uniform Trade Secrets Act (“WVUTSA”) defines a trade secret as

information, including, but not limited to, a formula, pattern, compilation, program, device, method, technique or process, that:

- (1) Derives independent economic value, actual or potential, from not being generally known to, and not being readily ascertainable by proper means by, other persons

who can obtain economic value from its disclosure or use;  
and

(2) Is the subject of efforts that are reasonable under the  
circumstances to maintain its secrecy.

W. Va. Code § 47-22-1(d). In *State ex rel. Johnson v. Tsapis*, 419 S.E.2d 1, 1 (W. Va. 1992), the Supreme Court of Appeals of West Virginia adopted the six-factor test found in § 757 of the Restatement (First) of Torts to determine whether there was good cause to issue a protective order to prevent the disclosure of the defendant's alleged trade secrets. In an unpublished opinion, the Fourth Circuit expressed its belief that the West Virginia Supreme Court of Appeals would also apply that six-factor test to determine the existence of a trade secret. *IVS Hydro, Inc. v. Robinson*, 93 F. App'x 521, 527 (4th Cir. 2004).

The *Tsapis* factors are as follows:

- (1) the extent to which the information is known outside of the [plaintiff's] business;
- (2) the extent to which it is known by employees and others involved in the [plaintiff's] business;
- (3) the extent of the measures taken by the [plaintiff] to guard the secrecy of the information;
- (4) the value of the information to the [plaintiff] and competitors;
- (5) the amount of effort or money expended by the [plaintiff] in developing the information; and
- (6) the ease or difficulty with which the information could be properly acquired or duplicated by others.

*Tsapis*, 419 S.E.2d at 3; *see also IVS Hydro*, 93 F. App'x at 526–27; *McGough*, 496 F. Supp. 2d at 739. “[T]he absence of evidence as to any one factor does not necessarily

preclude a finding that a trade secret exists. The factors should be viewed as instructive guidelines, weighed together in making that determination.” *McGough*, 496 F. Supp. 2d at 740 (citation omitted) (citing *Learning Curve Toys, Inc. v. PlayWood Toys, Inc.*, 342 F.3d 714, 722 (7th Cir. 2003)).

“[T]he hallmark of a trade secret is not its novelty but its secrecy.” *Avtec Sys., Inc. v. Peiffer*, 21 F.3d 568, 575 (4th Cir. 1994). “Absolute secrecy is not essential” to finding that a trade secret existed, and limited disclosure does not destroy secrecy. *Trandes Corp. v. Guy F. Atkinson Co.*, 996 F.2d 655, 664 (4th Cir. 1993). The Restatement (First) of Torts advises that to be a trade secret, “a substantial element of secrecy must exist, so that, except by the use of improper means, there would be difficulty in acquiring the information.” Restatement (First) of Torts § 757 cmt. b.

Misappropriation occurs when a trade secret is acquired by improper means, or when it is disclosed or used by someone who knows or has reason to know that the trade secret was “[a]cquired under circumstances giving rise to a duty to maintain its secrecy or limit its use.” W. Va. Code § 47-22-1(b).

### **1. CSS’s Source Code and Related Claims**

CSS alleges that the Defendants misappropriated a variety of trade secrets related to its software programs by using the information to create CT’s competing software, as well as provide support services for clients that use CSS’s software. Specifically, CSS alleges that the following are trade secrets: the source code for its software programs, “a detailed knowledge of the workings of the program[s],” “what

county clerks need[,] and how the software operates to meet their needs.” Pl.’s Resp. Defs.’ Mot. Summ. J. 11–12.

CSS initially provided its source code to county clerks’ offices by installing the source code files on the clients’ servers so that, in the event that CSS went out of business, the clients’ work would not be disrupted. Hr’g Tr., Jan. 25, 2017, Auburn Direct, 35:12–25. CSS provided its source code to clients in a way that the files “were pretty much editable in Notepad or however else you needed to view them.” Test. & Ex. Submission by CSS, Inc. Ex. 1, at 182:20–24 [ECF No. 105-1] (“Lowers Dep.”). At some point after May 2014, this “bad business practice” was discontinued, and all of CSS’s source code has since been removed from the county servers. Mot. by Defs. for Leave to File Under Seal Ex. 4, at 60:13–18 [ECF No. 70-4] (“Auburn Dep.”); Hr’g Tr., Jan. 25, 2017, Auburn Direct, 39:18–19.

All West Virginia county servers are connected through WVNet, “the virtual private network that allows vendors to access clients [and] state-run computers.” Defs.’ Mem. Opp’n Pl.’s Mot. Prelim. Inj. Ex. 3, at 267:7–15. Any company with a contract with any West Virginia county is given access to all county servers. Hr’g Tr., Jan. 25, 2017, Auburn Direct, 31:1–9, 35:15–18. CSS’s source code was neither password protected nor encrypted when it was on the county servers. *See* Hr’g Tr., Feb. 23, 2017, D. Herrington Direct, 120:21–23. In addition, CSS did not uniformly mandate any confidentiality requirements when it provided the county clients with its software and source code. *See* Hr’g Tr., Jan. 25, 2017, Auburn Direct, 32:12–24. CSS left decisions regarding who could access its source code on the county servers

entirely up to the clients themselves. *See id.* at 31:21–24. Thus, a competitor of CSS with access to WVNet may have been able to access CSS’s source code. *See* Hr’g Tr., Jan. 25, 2017, Auburn Direct, 31:1–20.

For approximately twenty-five years, CSS maintained the password “top gun” for users to access its software programs. Hr’g Tr., Feb. 23, 2017, D. Herrington Cross, 135:3–136:14. CSS did not enter into confidentiality agreements with the clerks’ offices that had the “top gun” password. *See id.* at 136:20–23. CSS disclosed the password to any county employees who attended a training session, and did not require the employees to keep the password confidential. *See id.* at 136:6–19.

Applying the *Tsapis* factors to these facts, “the extent of the measures taken by the [plaintiff] to guard the secrecy of” its source code weighs against finding that the source code constitutes a trade secret. CSS installed the source code files on county servers using its software for approximately twenty years. CSS did not encrypt or password protect the source code, nor did it label the source code as confidential. Finally, CSS did not mandate any confidentiality requirements when it provided its clients with the source code.

Additionally, considering “the ease . . . with which the information could be properly acquired or duplicated by others” also weighs against finding that the source code constitutes a trade secret. Any company with a contract with a West Virginia county, including competitors of CSS, could have accessed the source code through WVNet.

This shows that the source code was not “the subject of efforts that are reasonable under the circumstances to maintain its secrecy.” W. Va. Code § 47-22-1(d)(2). Certainly, placing the source code on the county servers without any requirement that it be kept confidential cannot be considered reasonable efforts at maintaining secrecy. Therefore, I **FIND** that CSS’s source code is not a trade secret.

Turning to CSS’s more abstract claims of trade secrets related to its source code—i.e., “a detailed knowledge of the workings of the program[s],” “what county clerks need[,] and how the software operates to meet their needs,”—I find that these also do not qualify as trade secrets under the WVUTSA. Given that a competitor of CSS could access the source code through proper means (i.e., through WVNet), a competitor with programming knowledge could then study the programs to obtain “a detailed knowledge of the workings of the program[s]” through proper means and without extraordinary effort.

As to “what county clerks need,” this cannot be a trade secret because of the extreme “ease . . . with which the information could be properly acquired or duplicated by others.” *Tsapis*, 419 S.E.2d at 3. Namely, any competitor who wished to acquire such information could simply call the county clerks’ offices and inquire about their software needs. Finally, combining this information with a knowledge of the software, a competitor could easily learn “how the software operates to meet [the county clerks’] needs.”

Thus, none of these types of information constitute trade secrets in this case because they do not satisfy the statutory requirement of “not being readily

ascertainable by proper means by other persons who can obtain economic value from their disclosure or use.” W. Va. Code § 47-22-1(d)(1). Therefore, I **FIND** that these are not trade secrets.

## 2. CSS’s Business Practices

The Defendants argue that they are entitled to summary judgment with respect to CSS’s trade secret claims concerning certain business practices. Specifically, the Defendants argue that the following types of information do not constitute trade secrets in this case: CSS’s plan to expand its business into Ohio; client lists and customer preferences; and CSS’s pricing methodology. Defs.’ Mem. Supp. Mot. Summ. J. 9–11. CSS has not responded to these arguments in its briefing, and I previously found that none of this information qualifies as a trade secret under the WVUTSA. *See* Prelim. Inj. Op. 36–40. Accordingly, I again **FIND** that CSS’s business-practices-related information does not constitute a trade secret.

## 3. Conclusion

I **FIND** that none of the information that CSS alleges to be a trade secret qualifies as a trade secret under the WVUTSA. Because I have found that no trade secret existed, I need not consider whether the Defendants “misappropriated” such information. I **FIND** that there is no genuine dispute of any material fact on CSS’s trade secret claim, and that the Defendants are entitled to judgment as a matter of law. Accordingly, the Defendants’ Motion for Summary Judgment is **GRANTED** with respect to **Count VI** (Misappropriation of Trade Secrets) of the First Amended Complaint.

### C. Breach of Contract

CSS alleges that Christopher Herrington breached the Confidentiality Agreement that he signed when he returned to CSS in September 2014 by using confidential information learned during his employment to 1) develop his competing software, and 2) provide support services to customers using CSS's software. The Confidentiality Agreement defines "Confidential and Proprietary Information" as

any and all data and information which is: (1) disclosed to or known by Employee as a consequence of, during, or through Employee's association with the Company; (2) not generally known outside the Company; and (3) which relates to any aspect of the Company or its business, research or development. Confidential and Proprietary Information includes, but is not limited to: research, technology, strategic plans, business plans, information relating to the Company's operating plans and methodologies, operations manuals, pricing and marketing strategies, financial information, or other trade secrets, as well as any or all information which is disclosed to Employee or in any way acquired by Employee relating to the Company or its employees, clients, customers, prospective customers or clients, suppliers, or vendors. Confidential and Proprietary Information may be written, verbal or recorded by electronic, magnetic or other methods, whether or not expressly identified as "Confidential" by the Company.

Pl.'s Mot. Prelim. Inj. Ex. 2, at ¶ 1.a. [ECF No. 65-2] ("Confidentiality Agreement").

The Agreement further provides,

During Employee's employment with the Company and thereafter, Employee will not copy, publish, convey, disclose nor use, directly or indirectly, for Employee's own benefit or for the benefit of any other person or entity (except the Company) any Confidential and Proprietary Information or otherwise utilize any Confidential and Proprietary Information for any purpose except in the course of work for the Company.



*Id.* at ¶ 2. The Agreement also contains a provision that the “Employee represents and warrants that Employee has not used or disclosed any Confidential Information Employee may have obtained from Employer prior to signing” the Agreement. *Id.* at ¶ 4.

### 1. Choice of Law

Before the court can assess the enforceability of the Confidentiality Agreement, I must determine which state’s law to apply. Because this court is sitting in diversity in West Virginia, I will apply West Virginia’s choice-of-law principles to determine which state’s law to apply to the contract. *See Klaxon Co. v. Stentor Electric Mfg. Co.*, 313 U.S. 487, 496 (1941).

The Confidentiality Agreement contains a choice-of-law provision that identifies Texas law as controlling. Confidentiality Agreement ¶ 7. However, no signatory to the Confidentiality Agreement resides in Texas, the parties signed the Agreement in West Virginia, and the alleged breach took place in West Virginia.

West Virginia’s conflict-of-laws jurisprudence provides “that a choice of law provision in a contract will not be given effect when the contract bears no substantial relationship with the jurisdiction whose laws the parties have chosen to govern the agreement[.]” *Gen. Elec. Co. v. Keyser*, 275 S.E.2d 289, 297 (W. Va. 1981). Under this standard, the parties and the court agree that West Virginia law governs the court’s interpretation of the Confidentiality Agreement. Therefore, I will apply West Virginia law to the breach of contract claim.

## 2. Enforceability of the Confidentiality Agreement

In my earlier opinion denying CSS's Motion for a Preliminary Injunction, I found that the Confidentiality Agreement was not unreasonable on its face. However, I found that CSS was not likely to succeed on the merits of its breach of contract claim because there was insufficient evidence that the Defendants had breached the Confidentiality Agreement. *See* Prelim. Inj. Op. 43–44. After carefully reviewing the Confidentiality Agreement and the applicable law, it is apparent that a more thorough analysis of the enforceability of the Agreement is necessary.<sup>4</sup>

The West Virginia Supreme Court of Appeals has not squarely addressed the enforceability of a nondisclosure agreement. Nevertheless, I have previously held that nondisclosure agreements should be analyzed in the same way as noncompete agreements. *See McGough*, 496 F. Supp. 2d at 752 (“Restrictive covenants include both noncompetition, nondisclosure and trade secrets covenants.”). “Covenants not to disclose and utilize confidential business information are related to general covenants not to compete because of the similar employer interest in maintaining competitive advantage.” *Id.* at 756.

First, the court must “look at the nondisclosure covenants to determine whether they are unreasonable on their face.” *Id.* at 755.

---

<sup>4</sup> Admittedly, the court erred by not conducting a more thorough analysis of the enforceability of the Confidentiality Agreement in its earlier opinion. At that stage of the litigation, where the plaintiff was seeking a preliminary injunction against the Defendants, the court was more concerned with whether CSS had produced enough evidence of breach to show that it was likely to succeed on the merits than it was with the legal enforceability of the Agreement. Now, where the Defendants seek summary judgment, CSS is only required to produce more than a mere “scintilla of evidence” to defeat the motion. *Anderson*, 477 U.S. at 252. Thus, determining the legal enforceability of the Confidentiality Agreement is imperative.

In deciding whether restraints on disclosure are reasonable two factors are important: (1) whether the employer is attempting to protect confidential information related to the business, such as trade secrets, methods of operation, names of customers, and personnel data even though the information does not rise to the [status] of a trade secret, and (2) whether the restraint is reasonably related to the protection of the information.

*Id.* at 756 (citing Restatement (First) of Contracts §§ 515(a), 516). “An excessively broad covenant with respect to time or geographic scope is unreasonable on its face.”

*Id.* at 752. “If the covenant is unreasonable on its face, it is utterly void and unenforceable.” *Id.*

Here, the Confidentiality Agreement contains no limitation with respect to time or geographic scope. Moreover, the Agreement purports to cover virtually all information learned by the employee during his employment. The only limitations are that the information is not generally known outside the company, and that the information relates to some aspect of the company (which is hardly a limitation at all). *See id.* at 756 (“By defining confidential information as essentially all of the information provided to Mr. McGough during his employment, the nondisclosure covenants amount to a post-employment covenant not to compete that is completely unrestricted in duration or geographic scope.”). In terms of conduct, the Confidentiality Agreement prohibits the employee not only from disclosing such information, but also from using it, “directly or indirectly,” except for the benefit of CSS. *See id.* (“The provisions are written so broadly as to cover everything Mr. McGough might have learned while working at Nalco[.] [I]f he were to strictly abide by its terms, he would be unable to ever work in a similar field again.”).

Given the extraordinarily broad language of the Confidentiality Agreement and absence of any limitation with respect to time or geographic scope, I **FIND** that “the restraint is [not] reasonably related to the protection of the information.” *Id.* Accordingly, I **FIND** that the Confidentiality Agreement is unreasonable on its face and therefore unenforceable.

### 3. Conclusion

I **FIND** that the Confidentiality Agreement, which serves as the basis of CSS’s breach of contract claim, is unreasonable on its face and therefore unenforceable. I **FIND** that there is no genuine dispute as to any material fact on CSS’s breach of contract claim, and that the Defendants are entitled to judgment as a matter of law. Accordingly, the Defendants’ Motion for Summary Judgment is **GRANTED** with respect to **Count II** (Breach of Contract) of the First Amended Complaint.

#### D. Unjust Enrichment

In the First Amended Complaint, CSS alleges that the “Defendants provided no consideration to CSS for their unlawful use and disclosure of the Confidential Information [and,] [a]s a result, Defendants have been unjustly enriched by their unlawful use and disclosure of the Confidential Information.” First Am. Compl. ¶¶ 57–58. CSS also alleges that the “Defendants’ unlawful and unjust use and disclosure of the Confidential Information has proximately caused damage to CSS.” *Id.* at ¶ 59.

The parties did not address the unjust enrichment claim in their briefing on the Defendants’ Motion for Summary Judgment. Nonetheless, this court has the authority “to enter summary judgments *sua sponte*, so long as the losing party was

on notice that she had to come forward with all of her evidence.” *Celotex Corp.*, 477 U.S. at 326. Although the Defendants did not specifically address the unjust enrichment claim in their briefing, their Motion for Summary Judgment requests that the court grant “Summary Judgment on all of Plaintiff’s claims.” Defs.’ Mot. Summ. J. 1. In addition, the Defendants’ Memorandum in Support of its Motion asserts that “this case should be dismissed in its entirety.” Defs.’ Mem. Supp. Mot. Summ. J. 20. Therefore, CSS was on notice to come forward with all of its evidence relevant to the unjust enrichment claim. *See Penley v. McDowell Cty. Bd. of Educ.*, 876 F.3d 646, 661 (4th Cir. 2017) (“Here, Gillespie moved for summary judgment on ‘all claims against him.’ . . . Accordingly, Appellant was on notice ‘to come forward with all . . . evidence’ relevant to his tortious interference claim, satisfying *Celotex*”).

“Unjust enrichment of a person occurs when he has and retains money or benefits which in justice and equity belong to another.” *Bright v. QSP, Inc.*, 20 F.3d 1300, 1306 (4th Cir. 1994) (quoting *Dunlap v. Hinkle*, 317 S.E.2d 508, 512 n.2 (W. Va. 1984)). Under West Virginia law, “[t]he elements of an unjust enrichment claim are: (1) a benefit conferred upon the defendant, (2) an appreciation or knowledge by the defendant of such benefit, and (3) the acceptance or retention by the defendant of the benefit under such circumstances as to make it inequitable for the defendant to retain the benefit without payment of its value.” *Emp’r Teamsters—Local Nos. 175/505 Health & Welfare Tr. Fund v. Bristol Myers Squibb Co.*, 969 F. Supp. 2d 463, 471 (S.D. W. Va. 2013) (alteration and internal quotation marks omitted). “The benefit may be an interest in money, land, chattels, or choses in action; beneficial services

conferred; satisfaction of debt or duty owed by him; or anything which adds to his security or advantage.” *Dunlap*, 317 S.E.2d at 512 n.2. “[I]f benefits have been received and retained under such circumstance that it would be inequitable and unconscionable to permit the party receiving them to avoid payment therefor, the law requires the party receiving the benefits to pay their reasonable value.” *Realmark Devs., Inc. v. Ranson*, 542 S.E.2d 880, 884–85 (W. Va. 2000).

Here, CSS has failed to identify the alleged benefit conferred upon the Defendants, the retention of which, without payment to CSS, would be inequitable or unconscionable. Instead, CSS merely states that the “Defendants provided no consideration to CSS for their unlawful use and disclosure of the Confidential Information.” First Am. Compl. ¶ 57. It is unclear to the court how this “unlawful use and disclosure of the Confidential Information” is meant to constitute the alleged benefit conferred upon the Defendants. Furthermore, even if the unlawful use and disclosure of confidential information could constitute a benefit, CSS has also failed to explain why the retention of this alleged benefit without payment to CSS would be inequitable under the circumstances. *See Emp’r Teamsters*, 969 F. Supp. 2d at 472 (dismissing unjust enrichment claim where plaintiff failed to “allege how Defendants’ retention of payments for a product that was effective in its ordinary purpose . . . rises to the level of constituting unjust enrichment”).

Because CSS has failed to identify the benefit conferred upon the Defendants, as well as the circumstances that would make the retention of any alleged benefit inequitable without payment to CSS, I **FIND** that CSS has not presented even a mere

scintilla of evidence to support its unjust enrichment claim. I also **FIND** that there is no genuine dispute as to any material fact, and that the Defendants are entitled to judgment as a matter of law. Accordingly, the Defendants' Motion for Summary Judgment is **GRANTED** with respect to **Count V** (Unjust Enrichment) of the First Amended Complaint.

#### **E. Duty of Loyalty**

CSS alleges that Christopher Herrington breached his duty of loyalty by creating his competing software and soliciting CSS's customers while still employed by CSS. First Am. Compl. ¶¶ 47–50. The Defendants assert that they are entitled to summary judgment on the duty of loyalty claim because it is “unclear” whether this is a standalone cause of action and, even if it is, it is “subsumed by” CSS's breach of contract and misappropriation of trade secrets claims. Defs.' Mem. Supp. Mot. Summ. J. 17.

The precise contours of the duty of loyalty in West Virginia are not entirely clear, and there is very little case law discussing this tort. However, what is clear is that West Virginia does recognize the breach of the duty of loyalty as an independent cause of action. *See Timberline Four Seasons Resort Mgmt. Co. v. Herlan*, 679 S.E.2d 329, 338 (W. Va. 2009) (“[B]ecause the specific facts of this case indicate that an agency relationship existed between the parties, we find that Ms. Herlan owed a duty of loyalty to her principal . . . Ms. Herlan, as an agent, cannot rightfully use her position to engage in self-dealing at the expense of her principal[.]”).

As the West Virginia courts often do, I will look to the Restatement for guidance concerning the elements of this claim.

(a) Except as otherwise provided in subsections (b) and (c), an employee breaches the duty of loyalty to the employer if, without the employer's express or implied consent, the employee, while employed by the employer, works for a competitor or otherwise competes with the employer.

(b) Competition with the employer under subsection (a) includes solicitation of the employer's customers to divert their business to a competitor and recruitment of other employees to work for a competitor, but does not include reasonable preparation by an employee or group of employees to compete with the employer.

Restatement of Employment Law § 8.04.

In this case, CSS alleges that Christopher Herrington used CSS's confidential information to create his competing software and solicit CSS's customers while he was still employed with CSS. The Defendants do not respond to these factual allegations. Therefore, drawing all reasonable inferences in favor of the nonmoving party (CSS), I **FIND** that there is a genuine dispute as to material facts relevant to the duty of loyalty claim. Accordingly, the Defendants' Motion for Summary Judgment is **DENIED** with respect to **Count III** (Duty of Loyalty) of the First Amended Complaint.

#### **F. Tortious Interference**

CSS alleges that the "Defendants willfully and intentionally interfered with" its contracts and relationships with its customers by using CSS's confidential information to market CT's software and services. First Am. Compl. ¶ 55. CSS further



alleges that the Defendants' interference "has actually and proximately damaged CSS." *Id.*

To establish prima facie proof of tortious interference, a plaintiff must show:

- (1) existence of a contractual or business relationship or expectancy;
- (2) an intentional act of interference by a party outside that relationship or expectancy;
- (3) proof that the interference caused the harm sustained; and
- (4) damages.

If a plaintiff makes a prima facie case, a defendant may prove justification or privilege, affirmative defenses. Defendants are not liable for interference that is negligent rather than intentional, or if they show defenses of legitimate competition between plaintiff and themselves, their financial interest in the induced party's business, their responsibility for another's welfare, their intention to influence another's business policies in which they have an interest, their giving of honest, truthful requested advice, or other factors that show the interference was proper.

*Tiernan v. Charleston Area Med. Ctr., Inc.*, 506 S.E.2d 578, 591–92 (W. Va. 1998) (quoting *Torbett v. Wheeling Dollar Sav. & Tr. Co.*, 314 S.E.2d 166, 167 (W. Va. 1983)).

CSS's lack of specificity in alleging the intentional acts of the Defendants that interfered with CSS's business relationships makes me doubt the viability of its tortious interference claim. However, viewing the evidence in the light most favorable to the nonmoving party (CSS), I **FIND** that there are genuine disputes of material facts relevant to this claim. Namely, I **FIND** that there is a dispute as to whether the

Defendants used improper means to interfere with CSS's contracts and relationships with customers, or whether the Defendants can show the defense of legitimate competition. In addition, I **FIND** that there is a dispute as to whether the Defendants' alleged interference caused the harm allegedly sustained by CSS. As the Defendants themselves point out, there is some evidence to suggest that some of CSS's customers "decided to switch to CT not because of any alleged 'tortious interference' of Defendants but because of their 'frustration with [CSS's] poor service.'" Defs.' Mem. Supp. Mot. Summ. J. 18. Accordingly, the Defendants' Motion for Summary Judgment is **DENIED** with respect to **Count IV** (Tortious Interference) of the First Amended Complaint.

#### **G. Declaratory Judgment ("Work for Hire")**

Finally, CSS requests a declaratory judgment that, under the "work for hire" doctrine, it is the author of any computer programs developed by Christopher Herrington during his employment with CSS. Although the parties did not address this claim in their briefing on the Defendants' Motion for Summary Judgment, CSS was on notice that it had to come forward with all of its evidence in response to the Defendants' request for summary judgment "on all claims." *See Celotex Corp.*, 477 U.S. at 326. Therefore, I will address this claim *sua sponte*.

The Copyright Act defines a "work made for hire" as "a work prepared by an employee within the scope of his or her employment." 17 U.S.C. § 101.<sup>5</sup> "In the case of a work made for hire, the employer . . . is considered the author for purposes of this

---

<sup>5</sup> Subsection (2) of the definition of "work made for hire" deals with specially commissioned works, and is not relevant to the facts of this case.

title, and, unless the parties have expressly agreed otherwise in a written instrument signed by them, owns all of the rights comprised in the copyright.” 17 U.S.C. 201(b).

In other words,

an employer owns a copyright in a work if (1) the work satisfies the generally applicable requirements for copyrightability set forth in 17 U.S.C. § 102(a), (2) the work was prepared by an employee, (3) the work was prepared within the scope of the employee’s employment, and (4) the parties have not expressly agreed otherwise in a signed, written instrument.

*Balt. Orioles v. Major League Baseball Players Ass’n*, 805 F.2d 663, 667 (7th Cir. 1986); *see also Miller v. CP Chems., Inc.*, 808 F.Supp. 1238, 1242 (D.S.C. 1992) (citing *Balt. Orioles*).

Here, the first and fourth elements (copyrightability and absence of a signed writing) are not at issue. Furthermore, there is no genuine dispute that Christopher Herrington was an employee of CSS, although there may be some dispute as to exactly how much of his competing software programs were developed during his employment rather than after his resignation from CSS. For purposes of this analysis, I will assume that the entirety of Christopher Herrington’s competing software programs were developed while he was still employed with CSS.

The real issue here is whether Christopher Herrington’s competing software programs were “prepared within the scope of the employee’s employment.” The Supreme Court has made clear “that common-law agency principles govern resolution of that question.” *Avtec Sys.*, 21 F.3d at 571 (citing *Cnty. for Creative Non-Violence v. Reid*, 490 U.S. 730, 739–40 (1989)). “[A] servant’s conduct is within the scope of employment ‘only if: (a) it is of the kind he is employed to perform; (b) it occurs

substantially within the authorized time and space limits; [and] (c) it is actuated, at least in part, by a purpose to serve the master.” *Id.* (second alteration in original) (quoting Restatement (Second) of Agency § 228). Again, there is no dispute that Christopher Herrington’s software programs are “of the kind” of work that he was employed to perform. And, while there may be a dispute as to whether Christopher Herrington developed his programs on CSS’s time, or using their facilities, I will assume for the sake of this analysis that this element is satisfied.

However, it is clear that the development of Christopher Herrington’s software programs was not “actuated, at least in part, by a purpose to serve” CSS. In order to satisfy this condition, CSS must show that Herrington “was at least appreciably motivated by a desire to further [CSS’s] corporate goals.” *Avtec Sys.*, 21 F.3d at 572 (alteration and internal quotation marks omitted); *see also* Restatement (Second) of Agency § 235 (“An act of a servant is not within the scope of employment if it is done with *no intention* to perform it as a part of or incident to a service on account of which he is employed.” (emphasis added)). Clearly, Christopher Herrington’s purpose in designing his software programs was to compete with CSS, not to further their corporate goals. Thus, Herrington developed his competing programs with *no intention* to serve his employer, CSS.

Therefore, I **FIND** that Christopher Herrington’s computer programs were not “prepared within the scope of [Herrington’s] employment” with CSS. The programs are not “works made for hire” within the meaning of the Copyright Act, and CSS is not considered the author of the programs for copyright purposes. I also **FIND** that

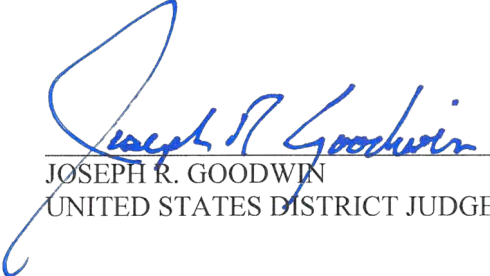
there is no genuine dispute of material fact relevant to the declaratory judgment claim,<sup>6</sup> and that the Defendants are entitled to judgment as a matter of law. Accordingly, the Defendants' Motion for Summary Judgment is **GRANTED** with respect to **Count VIII** (Declaratory Judgment) of the First Amended Complaint.

#### **IV. Conclusion**

In summary, the Defendants' Motion for Summary Judgment [ECF No. 123] is **GRANTED** with respect to **Count I** (Copyright Infringement), **Count II** (Breach of Contract), **Count V** (Unjust Enrichment), **Count VI** (Misappropriation of Trade Secrets), and **Count VIII** (Declaratory Judgment) of the First Amended Complaint. The Defendants' Motion for Summary Judgment is **DENIED** with respect to **Count III** (Duty of Loyalty), **Count IV** (Tortious Interference), and **Count VII** (Injunctive Relief) of the First Amended Complaint.

The court **DIRECTS** the Clerk to send a copy of this Order to counsel of record and any unrepresented party.

ENTER: January 29, 2018

  
JOSEPH R. GOODWIN  
UNITED STATES DISTRICT JUDGE

---

<sup>6</sup> Although I indicated that there may be some disputes of fact related to the declaratory judgment claim, these facts are immaterial because I have found that CSS is unable to establish an essential element of the work-for-hire test. Regarding that essential element (i.e., whether Christopher Herrington's computer programs were prepared within the scope of his employment), there is no genuine dispute of any material fact.