

DEMOCRATIZING KNOWLEDGE

— through —

PRINT

Visual Communications Journal

Fall 2016, Volume 52, Number 2

Mobile Application Development: Native vs Web-based vs Hybrid

TAYLOR YOUNGBLOOD for Professor JERRY WAITE, Ed.D.





GRAPHIC
COMMUNICATIONS
EDUCATION
ASSOCIATION

Visual Communications JOURNAL

FALL 2016

Volume 52 Number 2

Acknowledgements

Editor

Dan Wilson, Illinois State University

Editorial Review Board

Cynthia Carlton-Thompson, North Carolina A&T State University
Bob Chung, Rochester Institute of Technology
Christopher Lantz, Western Illinois University
Devang Mehta, North Carolina A&T State University
Tom Schildgen, Arizona State University
Mark Snyder, Millersville University
James Tenorio, University of Wisconsin–Stout
Renmei Xu, Ball State University

Cover Design

Ben Alberti, Western Technical College
Instructor, Barbara Fischer

Page Design, Layout, and Prepress

Janet Oglesby and Can Le

Printing, Bindery, and Distribution

Harold Halliday, University of Houston
University of Houston Printing and Postal Services

About the Journal

The *Visual Communications Journal* serves as the official journal of the Graphic Communications Education Association, and provides a professional communicative link for educators and industry personnel associated with design, presentation, management, and reproduction of graphic forms of communication. Manuscripts submitted for publication are subject to peer review. The views and opinions expressed herein are those of authors and do not necessarily reflect the policy or the views of the GCEA.

Article Submission

Please follow the guidelines provided at the back of this *Journal*.

Membership and Subscription Information

Information about membership in the Association or subscription to the *Journal* should be directed to the GCEA First Vice-President.

Reference Sources

The *Visual Communications Journal* can be found on EBSCOHost databases.
ISSN: Print: 0507-1658 Web: 2155-2428

President – Mike Stinnett

Royal Oak High School (Ret.)
21800 Morley Ave. Apt 517
Dearborn, MI 48124
(313) 605-5904
president@gceaonline.org

President-Elect – Malcolm Keif

Cal Poly University
Graphic Communications
San Luis Obispo, CA 93407
805-756-2500
presidentelect@gceaonline.org

First Vice-President (Publications)

Gabe Grant

Eastern Illinois University
School of Technology
600 Lincoln Avenue
Charleston, IL 61920
(217) 581-3372
firstvp@gceaonline.org

Second Vice-President (Membership)

Can Le

University of Houston
312 Technology Bldg.
Houston, TX 77204-4023
(713) 743-4082
secondvp@gceaonline.org

Secretary – Laura Roberts

Mattoon High School
2521 Walnut Avenue
Mattoon, IL 61938
(217) 238-7785
secretary@gceaonline.org

Treasurer – Pradeep Mishra

Arkansas State University
P.O. Box 1930
State University, AR 72467
(870) 972-3114
treasurer@gceaonline.org

Immediate Past President – Tom Loch

Harper College
Graphic Arts Technology
1200 W. Algonquin Road
Palatine, IL 60067
(847) 255-7142
pastpresident@gceaonline.org

Mobile Application Development: Native vs Web-based vs Hybrid

by Taylor Youngblood for Professor Jerry Waite, Ed.D. • University of Houston

Abstract

Research was conducted in order to discern the most efficient method for producing interactive mobile applications, according to professionals in the greater Houston area. This research aimed to provide an insight into the preferences of Houston production agencies and current coding requirements for mobile application development. The research data for this study was collected via a Google survey that was emailed to willing participants. Employees from 30 Houston mobile application production agencies were contacted and 13 participants anonymously responded. Data from these responses were collected and indicated that developers prefer and most often develop web-based applications. These developers indicated that aspiring developers should have a working knowledge of HTML5, CSS3, JavaScript, Objective C, JQuery, C#, PHP, Rails, or Kotlin in order to be considered an asset to a mobile application production firm.

Mobile Application Development: Native vs Web-based vs Hybrid

The newest coined phrase in the technological realm is “there’s an app for that.” Applications have taken over the mobile development world and have become an ever increasing market for graphic designers and website developers. Since demand has increased, production houses must constantly search for efficient ways to design, develop, and produce mobile applications. Even though there are several avenues for developing mobile applications, production houses tend to invest time and energy into complete understanding of one method and continue utilizing it because of familiarity. Often these production houses fail to consider software being utilized by similar professionals. This study was conducted to determine the most effective avenue for developing interactive mobile applications.

In the mobile development industry, there are several generally accepted methods utilized to develop interactive mobile applications. These methods include native applications, web applications, and hybrid applications such as Phonegap, Aquafadas, and Appcelerator. Each method has distinct advantages and disadvantages. User preference, functionality, coding requirements, and application diversification are measurements that can be used to

discern the best approach for mobile application development. Since several avenues for application development exist, the pertinent question is: according to mobile application development professionals in the greater Houston area, what is the most efficient method for producing interactive mobile applications?

Since the iPhone was first released in 2007, smartphones, tablets, and e-readers have evolved from luxury items used only by elite business professionals to a fundamental household device. Advances in technology, and more efficient application production methods, have led to a noteworthy increase in the number of mobile application users. This is significant for publishing houses and corporations because “mobile phones now account for 62% of all time spent online” (Seetharaman, 2015, p.2) with 70% of that time in applications. Just as production houses and corporations must keep abreast of new mobile application advances, college students emerging with four-year degrees must also be aware of new technological trends. Since mobile application development is a field that is experiencing immeasurable growth, college graduates would be wise to know best practices for development of these applications.

Throughout this paper, a number of terms will appear repeatedly. The meaning and intention of each term is clearly defined below.

- Interactive mobile application is defined by *PC Mag Encyclopedia* as a “software application that runs in a smartphone, tablet, or other portable device” (“Mobile App Definition from *PC Magazine Encyclopedia*,” n.d.).
- Application programming interface (API) is “language and message format used by an application program to communicate” (“API Definition from *PC Magazine Encyclopedia*,” n.d.) with a parent application.
- Native application is a “mobile app programed in the recommended language for the mobile operating system; for example, Objective C or Swift for iOS and Java for Android” (“Native Mobile App Definition from *PC Magazine Encyclopedia*,” n.d.).
- Web application as “an application in which all or some parts of the software are downloaded from the

Web each time it is run” (“Web Application Definition from *PC Magazine Encyclopedia*,” n.d.).

- Hybrid application is written utilizing a single source code where “any changes to the source generates new apps for all the targeted platforms” (“Hybrid App Definition from *PC Magazine Encyclopedia*,” n.d.). PhoneGap, Aquafadas, and Appcelerator are three types of hybrid applications.
- Phonegap is a framework that utilizes a library of code to convert a single programming language into device specific languages.
- Aquafadas uses either InDesign Authoring or Cloud Authoring to create interactive mobile applications.
- Appcelerator is a multi-platform mobile application development software program that allows developers to build fully native applications from a single JavaScript code (“Mobile App Development Platform – Appcelerator,” n.d.).

Literature Review

There are several different methods for developing a mobile application. Each method will be discussed individually in accordance with its advantages and disadvantages. Native applications use code that is “written specifically for a particular phone’s operating system” (Cha, 2013, p.151). Examples of coding languages utilized for the development of native applications are Objective-C for iPhone, Java for Android, and C# for Windows Mobile (Cha, 2013, p.151). Native apps are recommended for “hardware intensive applications” (Jobe, 2013, p.31) because its “code can access - APIs that can access device storage, sensors, and data” (Charland, 2011, p.4). Since this code is downloaded, it has the ability to run faster and more efficiently than other applications (Charland, 2011, p.7). The drawback to native applications is that they are expensive to create since a separate code must be developed for all distinct platforms (Charland, 2011, p.1). Also, since these applications have “unhindered access to device hardware” (Jobe, 2013, p.28) they can become a security risk (“Patents,” 2013, p.1). Developers and clients must consider application, time, and cost requirements before selecting this production method.

A second method for developing mobile apps is the use of web applications. Web applications are “designed and developed to mimic” (Jobe, 2013, p.28) native applications while running in a web browser. These apps utilize languages such as HTML5, JavaScript, CSS, and Ruby”

(Cha, 2013, pp.151–152). Open source coding allows web applications to be accessed and utilized across multiple mobile platforms (Jobe, 2013, p.28). Web applications tend to be less expensive, faster to develop, and reduce long term maintenance costs (Cha, 2013, p.152). One major shortcoming of web applications is their “lack of direct access to the device hardware,” which causes the application to appear “slower and more sluggish” (Jobe, 2013, p.31) than native applications. Web apps also have a tendency to become unstable and have a limited user interface that is dependent on the mobile platform (Cha, 2013, p.152). Customers may prefer this option because of low cost and faster development.

The third method for developing mobile applications is through the use of hybrid application software. Hybrid apps are “neither truly a mobile web app or a native app” (Jobe, 2013, p.29). These applications utilize HTML5, JavaScript APIs, or CSS and run “inside a 3rd party native app container” (Jobe, 2013, p.29). These native app containers typically give the hybrid applications access to “native device APIs and hardware” (Jobe, 2013, p.29). PhoneGap, Aquafadas, and Appcelerator are three currently popular hybrid frameworks.

PhoneGap is an open source framework that combines “HTML, CSS, and JavaScript with an API for tapping native device hardware and functionality” (Borck, 2012, p.1). These web-based applications can access “everything from the accelerometer to camera, compass, contacts, file, geolocation, media (audio and video), network, notifications (alert, sound, vibration) and storage” (Gibbs, 2012, p.18). Once an app has been built with PhoneGap software it can be accessed on “Android, iOS, Windows Phone, BlackBerry, Symbian, WebOS, and Bada” (Borck, 2012, p.1). PhoneGap is a free service that allows the developer to include features that resemble native applications such as “plug-ins to mimic native” (Borck, 2012, p.1) user interface, push notifications, and PayPal integration. Even though PhoneGap apps can appear native for data-driven web applications they are not the most efficient option for “processor- or graphics-intensive applications” (Borck, 2012, p.1). Any type of new code packing software requires developers to invest time in learning a new API. Before developers invest their time in a particular method they must be certain that there is a demand for hybrid applications.

Aquafadas is another cross-platform mobile application development software that specializes in the publication of “interactive digital content such as books, comics,

corporate documents, magazines, and newspapers” (Kobo and Aquafadas, 2014, p.2). These applications are built in two distinct steps. First, a publisher wanting to transform print media into digital media would upload all of their “PDF, JPEG, PNG and TIFF files” (Oskolkova, 2014, p.1) into Aquafadas’ Cloud Authoring or InDesign software program. Once the files are uploaded the publisher completes the metadata, organizes the page layout, and adds enrichments before the software converts the files into “ePUBs and Web Reader files” (Oskolkova, 2014, p.1). When the files are digitized they can be transferred to the AppFactory software where the developer can create native iOS or Android applications (“About Aquafadas,” n.d.). With “one click,” all of the digitized files can be coded for either platform and published to the “Apple, Google, or Amazon app stores” (“About Aquafadas,” n.d.). Since this software is cloud-based, the only requirement for access is an Internet connection (Kobo and Aquafadas, 2014, p.2). Although the software is free to download, it requires users to purchase licenses before publishing the applications (Oskolkova, 2014, p.3). Since Aquafadas is a specialized software used for the development of interactive digital content, developers should be certain that this program matches their clients’ needs before recommending this approach.

Finally, Appcelerator is a multi-platform mobile application development software program that allows developers to build fully native applications from a single JavaScript code (Mobile App Development Platform – Appcelerator, n.d.). Appcelerator allows clients to utilize a program called Arrow, which is a “framework for building and running mobile-optimized APIs” (Nicastro, 2015, p.1) or Titanium “for creating native, cross-platform applications” (Nicastro, 2015, p.3). Developers using the Appcelerator platform can build applications for any device or operating system, verify applications with a mobile test automation feature, access mobile-optimized data sources, and measure application success by detecting crashes, monitoring usage, and tuning performance (Mobile App Development Platform – Appcelerator, n.d.). In Appcelerator, tuning performance is painless because of Fastdev (Perez, 2011, p.2). Fastdev allows developers to “make changes to code in real-time while the app runs in the simulator” (Perez, 2011, p.2). Developers also have the ability to access a “full-featured source code editor, code completion, Git integration and debugging capabilities” (Perez, 2011, p.2) for a wide range of platforms. Since Appcelerator offers several application building tools it is not a free software. Developers must choose

from three levels of professional membership and pay the corresponding fee. Application developers must consider the monthly fees and the effect of that cost on their monthly overhead.

Method

In order to determine the most efficient method for developing a mobile application according to Houston professionals, a survey was sent to willing participants. Surveys were distributed to participants over the course of the Spring 2016 academic semester. Participants were selected based on their geographic location and specialization in mobile application development. Developers from 30 Houston application development companies were contacted and asked to anonymously complete a Google survey. The survey was also posted on the University of Houston’s Digital Media LinkedIn page so that alumni of the program will have the opportunity to complete the survey. A replica of the Google Survey can be found in Table 1.

Once all participant surveys were collected, the results were tallied and displayed in a series of pie charts. Finalized results indicated which development method customers commonly requested during their kickoff meetings with developers, which method developers preferred for mobile application development, and which method they used most frequently for client projects.

Results

The results of this study are based on the data provided anonymously by 13 participants. Even though 30 Houston mobile application development companies were contacted and the survey was posted on social media outlets, actual participation was limited to about 43% of the potential pool of respondents. Since the parameters of this study required participation from professionals with a specific skill set, the researcher anticipated that participation could be nominal. The data received was sufficient to draw some conclusions about mobile application development preferences among Houston professionals, but should not be extrapolated to other geographic areas.

Since mobile application developers would not have work without demand from customers, the first part of the research pertained to customer preferences. Participants were asked what type of application customers preferred to have developed upon arrival for their initial kick-off meeting with a development firm. Initially 62% of cus-

customers preferred native applications, 38% preferred web applications, and none of the customers knew about the possibility of hybrid applications. Developers stated that 92% of their customers did not have a firm grasp of the difference between native, web, and hybrid applications. A pie chart depicting these results can be found in Figure 1. During this meeting, developers were able to explain the options available to their customers along with the advantages and disadvantages of each option. Based on this provided information, a majority of customers decided to select a different mode of development.

What Customers Initially Prefer

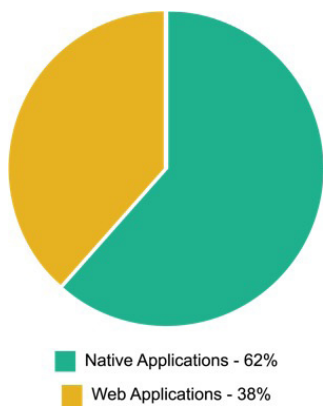


Figure 1

Another facet of the research focused on the personal preferences of developers in regards to the use of web, native, and hybrid applications. In this section of the survey, the results demonstrated that 54% of participants preferred to develop web applications, 15% preferred to develop native applications, and 31% preferred to develop hybrid applications. The results from this section of research can be found in Figure 2. Developers stated that their preferences stemmed from the parameters of the specific project, coding requirements, and application functionality.

Finally, participants were asked to report which type of application their company was contracted to develop most often. This is the type of application that customers selected after receiving all pertinent information about cost, functionality, design, and future maintenance. Participants reported that only 8% of clients selected native applications, 69% selected web applications, and 23% preferred hybrid applications. The results from this portion of research can be found in Figure 3. Customers gravitated toward different development modes after the conclusion of their kick-off meetings.

What Developers Prefer

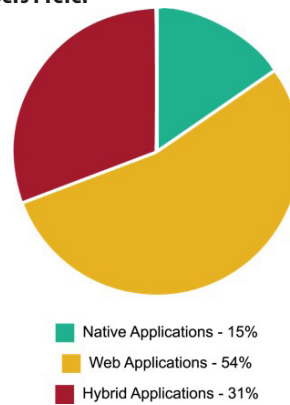


Figure 2

What Is Actually Developed

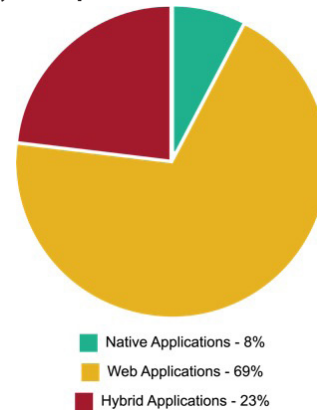


Figure 3

Conclusions

From the results of this research, conclusions can be drawn about uniformed client preferences, functionality, coding requirements, and application development. Uninformed clients arrived for their kick-off meetings with developers not truly understanding the options available for mobile application development or the benefits and drawbacks of each method. This means that developers must possess enough knowledge about each method to assess which mode of production is most appropriate for the customer's application requirements.

The surveyed developers reported that they preferred to develop web and hybrid applications because of their functionality. Of the developers surveyed, 54% stated that they preferred to create web applications because they are cross-platform compatible, do not require installed software, do not require manual updates, and allow consumers to constantly access current versions.

There was a significant number of participants who preferred developing hybrid applications. One participant wrote that they enjoyed having the ability to “wrap a web-based app and give it access to the native layer.” These developers also stated that another advantage of hybrid applications was the ability to easily make changes to the code without resubmitting the entire application through a store.

Finally, 15% of developers stated that they preferred native applications because their clientele tended to request hardware intensive applications that required the code to access specific APIs.

In accordance with functionality, participants were asked which application would be easier for customers to manage and update in the future. Of the developers surveyed, 69% stated that web applications are easiest for clients to manage and 31% stated that hybrid applications can be more easily updated. All of the developers agreed that clients would not have the ability to update native applications without coding knowledge. This section of research shows that Houston professionals believe that web applications are the most efficient method for developing mobile applications, followed by hybrid applications, and then finally native applications.

Participants were asked to list the coding languages that were required for professionals in their field. Participants noted that applications developers should be able to demonstrate proficiency in HTML5, CSS3, JavaScript, Objective C, JQuery, and C#. Each of these languages was listed and discussed throughout this research. In addition to these languages developers also stated that PHP, Rails, and Kotlin were necessary languages for the development of interactive mobile applications. Once coding languages were established the developers were asked if their companies required them to code in a specific language or if they were given the freedom to choose which coding language best suited each project. Every developer surveyed stated that their organization gave them the freedom to choose their own coding language. From this research, applicants searching to acquire positions with mobile application development agencies can discern that it would be wise to have a thorough knowledge of each aforementioned coding language.

Finally, participants were asked which method of mobile application development their company was most often contracted to develop. Participants reported that 69% most often developed web applications, 23% of them mostly developed hybrid applications, and 8% of them

developed native applications. This piece of research supports the conclusion that Houston mobile application development professionals believe that web-based applications are the most efficient mode for application development. Hybrid applications and native applications fall into second and third place, respectively, according to their overall development efficiency.

Recommendations

In order to solidify the results of this study, additional research must be conducted. This research should include more participants from varying geographic locations. Future researchers should also consider the cost and time associated with each method of application development, user experience, and client satisfaction. The results from this future research will provide further understanding into the preferences of developers and clients in regards to mobile application development. This research is important to the digital media industry because mobile applications are quickly becoming the most popular way for consumers to access information about businesses and products. This is a booming industry and research must stay abreast of this technological trend.

References

- About Aquafadas. (n.d.). Retrieved November 29, 2015, from <http://www.aquafadas.com>
- API Definition from *PC Magazine Encyclopedia*. (n.d.). Retrieved from <http://www.pcmag.com/encyclopedia/term/37856/api>
- Borck, J. R. (2012). PhoneGap: Mobile development made easy. InfoWorld.Com, Retrieved from <http://search.proquest.com.ezproxy.lib.uh.edu/docview/921520422?accountid=7107>
- Cha, S., & Yun, Y. (2013). Smartphone Application Development using HTML5- based Cross-Platform Framework. *ASTL*, 20, 151–153.
- Charland, A., & LeRoux, B. (2011, February 6). Mobile Application Development: Web vs. Native. *ACMQueue*.
- Gibbs, M. (2012). PhoneGap fills the smartphone development gap. *Network World*, 29(19), 18. Retrieved from <http://search.proquest.com.ezproxy.lib.uh.edu/docview/1152020952?accountid=7107>

- Hybrid App Definition from *PC Magazine Encyclopedia*. (n.d.). Retrieved from <http://www.pcmag.com/encyclopedia/term/66836/hybrid-mobile-app>
- Jobe, W. (2013). Native Apps vs. Mobile Web Apps. *International Journal of Interactive Mobile Technologies*, 7(4), 27-32. Retrieved November 29, 2015, from <http://online-journals.org/index.php/i-jim/index>
- Kobo and aquafadas collaborate to create innovative magazine experience. (2014, May 13). PR Newswire Retrieved from <http://search.proquest.com.ezproxy.lib.uh.edu/docview/1523860125?accountid=7107>
- Mobile App Definition from *PC Magazine Encyclopedia*. (n.d.). Retrieved November 29, 2015, from <http://www.pcmag.com/encyclopedia/term/60015/mobile-app>
- Mobile App Development Platform – Appcelerator. (n.d.). Retrieved November 29, 2015, from <http://www.appcelerator.com/>
- Native Mobile App Definition from *PC Magazine Encyclopedia*. (n.d.). Retrieved November 29, 2015, from <http://www.pcmag.com/encyclopedia/term/65705/native-mobile-app>
- Nicastro, D. (2015, April 2). Appcelerator Eyes Big Fish in Mobile App Dev Space. Retrieved November 30, 2015, from <http://www.cmswire.com/cms/mobile-enterprise/appcelerator-eyes-big-fish-in-mobile-app-dev-space-028675.php>
- Oskolkova, A., & Leger, L. (2014). Aquafadas Cloud Authoring Brings Digital Publishing to the Cloud.
- Patents; researchers submit patent application, “filesystem access for web applications and native code modules”, for approval. (2013). Politics & Government Week, , 9743. Retrieved from <http://search.proquest.com.ezproxy.lib.uh.edu/docview/1447186935?accountid=7107>
- Perez, S. (2011, June 13). Appcelerator Launches Titanium Studio: Mobile, Desktop & Web Development in One. Retrieved November 30, 2015, from <http://readwrite.com/2011/06/13/appcelerator-launches-titanium-studio-mobile-desktop-web-development-in-one>
- Seetharaman, D. (2015, September 22). Websites Attract Bigger Audiences, but Users Spend More Time in Apps. Retrieved from <http://blogs.wsj.com/digits/2015/09/22/websites-attract-bigger-audiences-but-users-spend-more-time-in-apps>
- Web Application Definition from *PC Magazine Encyclopedia*. (n.d.). Retrieved November 29, 2015, from <http://www.pcmag.com/encyclopedia/term/54272/web-application>

Manuscript Form and Style

- Prepare manuscripts according to the APA style, including the reference list.
- List your name and address on the first page only. Article text should begin on the second page.
- Provide a short biography for yourself that can be used if the article is accepted for publication.
- All articles must be submitted in electronic form on a CD-ROM or as an email attachment.
- Submit a Microsoft Word document, maximum of 10 pages (excluding figures, tables, illustrations, and photos). Do not submit documents created in page-layout programs.
- Word documents must have been proofread and be correct.
- Call out the approximate location of all tables and figures in the text. Use the default style “Normal” on these callouts. The call-outs will be removed by the designer.
- Use the default Word styles only. Our designer has set up the page layout program styles to correspond to those style names.
 - ◆ Heading 1
 - ◆ Heading 2
 - ◆ Heading 3
 - ◆ Normal

Graphics

- Be sure that submitted tables and other artwork are absolutely necessary for the article.
- Write a caption for each graphic, include captions in a list at the end of your Word document.
- Electronic artwork is preferred and should be in PDF or TIFF format.
- Send all artwork files and hard copies of these files with your submission.

Tables

- Set up tables in separate documents, one document for each table.
- Do not attempt to make it “pretty.” Use the default Word style “Normal” for all table text. Do not use any other formatting.

- Do not use hard returns inside the table (“enter” or “return”).
- Get the correct information into the correct cell and leave the formatting to the designer.
- Tables will be formatted by the designer to fit in one column (3.1667" wide) or across two columns (6.5" wide).

Artwork

- Scan photographs at 300 ppi resolution.
- Scan line drawings at 800 ppi resolution.
- Screen captures should be as large as possible.
- Graphics should be sized to fit in either one column or across two columns.
 - ◆ One column is 3.1667" wide, two columns are 6.5" wide.
 - ◆ Graphics may be larger than these dimensions, but must not be smaller.

Manuscript Guidelines

Eligibility for Publication

Members of the Graphic Communications Education Association, or students of GCEA members, may publish in the *Visual Communications Journal*.

Audience

Write articles for educators, students, graduates, industry representatives, and others interested in graphic arts, graphic communications, graphic design, commercial art, communications technology, visual communications, printing, photography, desktop publishing, or media arts. Present implications for the audience in the article.

Types of Articles

The *Visual Communications Journal* accepts four levels of articles for publication:

1. Edited articles are accepted or rejected by the editor. The editor makes changes to the article as necessary to improve readability and/or grammar. These articles are not submitted to a panel of jurors. The decision of the editor is final.
2. Juried articles are submitted to the editor and are distributed to jurors for acceptance/rejection. Juried articles are typically reviews of the literature, state-of-the-art technical articles, and other nonempirical papers. Jurors make comments to the author, and the author makes required changes. The decision of the jurors is final.
3. Refereed articles are submitted to the editor and are distributed to jurors for acceptance/rejection. Refereed articles are original empirical research. Jurors make comments to the author and the author makes required changes. The decision of the jurors is final.
4. Student articles are submitted by GCEA members and are accepted/rejected by the editor. These articles are not submitted to a panel of jurors. The editor's decision is final. Please be aware that poorly written student papers will be rejected or returned for editing.

Submittal of Manuscripts

All manuscripts must be received by the editor no later than December 15th to be considered for the spring *Journal* or by June 15th to be considered for the fall *Journal*. Include digital copies of all text and figures. Prepare text and artwork according to the instructions given in these guidelines. Be sure to include your name, mailing address, e-mail address, and daytime phone number with your materials. E-mail all materials to the editor (address shown below).

Acceptance and Publication

If your article is accepted for publication, you will be notified by e-mail. The *Visual Communications Journal* is published and distributed twice a year, in the spring and in the fall. Printed copies are mailed to GCEA members. A PDF version of the *Journal* is published online at www.GCEAonline.org.

Notice

Articles submitted to the *Journal* cannot be submitted to other publications while under review. Articles published in other copyrighted publications may not be submitted to the *Journal*, and articles published by the *Journal* may not be published in other publications without written permission of the *Journal*.

Submit All Articles and Correspondence to:
Dan Wilson, dan.wilson@illinoisstate.edu
or check www.GCEAonline.org for contact information
for the GCEA First Vice-President.

See following page for style guidelines