Mobile Website Creation based on Web Data eXtraction and Reuse

Chia-Hui Chang, Yan-Kai Lai, Yu-An Chou, Oviliani Yenty Yuliana

Department of Computer Science and Information Engineering
National Central University

Abstract—Due to the rapid growth of mobile devices, using mobile devices to access the Internet is more and more popular. However, many websites are designed for desktop browsing and do not have Responsive Web Design (RWD) for mobile devices. Lacking mobile version for websites make it difficult for users to operate on mobile devices and may decrease the exposure rate or lose some commercial possibilities. However, abandon existing websites to create an RWD website (such as Wix) is not always a good idea since many services (e.g. backend management functions) need to be operated in a desktop GUI. In this paper we introduce a project called DeXaR (Data eXtraction and Reuse) to enable users to quickly and easily create an RWD website from existing website without programming. By incorporating automatic Web data extraction techniques to support Web Data ETL (Extract-Transform-Load) services, we are able to link contents of an existing website to the new website. In other words, we can keep mobile website synchronized with the existing website via data APIs such that users only need to maintain one copy of the data but it can be reused in mobile webpages. The user study on 35 students shows the design philosophy for mobile website creation from existing website is encouraging. However, further survey on the linkage of web data extraction needs to be explored.

1. Introduction

Due to the booming of smart handheld devices, there is an urgent need to browse the Web in mobile devices since people need to obtain information no matter where they are or what they want. While people spend more time on handheld devices to browse the web, most websites are designed for the desktop computers only and do not support small device browsing. Therefore, users must constantly zoom in and shift the display of the textual content and functional buttons on the device. These repetitive operations will cause users great distress and poor browsing experience.

In order to solve the problem in browsing web pages on handheld devices, the concept of Responsive Web Design (RWD) was proposed to enable the websites to dynamically adapt its layout to fit the size and orientation of the device on which it is viewed. Through CSS media query, web developers can check the capabilities of the device and apply corresponding stylesheets or style rules to control the display on different devices. By assigning different style sheets for different media, handheld devices can display the same website normally to avoid repeated zooming, panning and scrolling operations.

While RWD has been introduced to alleviate the website browsing problem on mobile devices, many organizations such as middle schools and hospitals still do not have mobile-device friendly websites because of extra cost and lack of proper IT staff. Meanwhile, creating an RWD website requires more web technologies such as HTML5.0, CSS or JavaScript, Node.js, etc.

In this paper, we introduced the DeXaR project to quickly create an RWD website from existing websites. Similar to existing mobile web design systems such as Wix and Weebly, it requires no programming to create a RWD website. Users can quickly create an RWD website by selecting a predefined template or customizing a page with various web components with page preview. However, different from existing mobile website design systems, this project features the idea of synchronizing the created mobile website and original website based on web data extract-transform-load APIs such that the same backend management system could be used to maintain the website.

To achieve this goal, the DeXaR project also includes a web data ETL system for data API creation by integrating web scraping techniques and web data extraction technologies, where the former helps download web pages and the later outputs the data in a tabular spreadsheet. The web data ETL system allows users to specify the web URLs (universal resource locator) and can extract data from a single web page that contains a list or multiple web pages that are generated from the same template to generate a data API endpoint. Thus, users can use the extracted data in existing web pages as input for mobile web page creation via the mobile web creator. This is especially helpful when there are list items since users do not need to input items one by one.

2. Related Work

Transforming desktop websites for better browsing experience on handheld devices has been a research topic in Web programming research. For example, Mohan et al. (1999) introduced InfoPyramid which provides a multimedia content description framework in a multi-abstraction, multi-modal content representation for adaptive delivery based on profile of client device. Xiao et al. (2009) also proposed a Web page transformation method by slicing the original page into page blocks iteratively to build a tree hierarchy of the transformed pages.

In addition, Nichols et al. (2008) introduced the Highlight system for creating and deploying mobile Web applications via a proxy server to clip and transform content from existing web pages for the mobile device. Koehl and Wang, enhanced the idea

Contact: Chia-Hui Chang, National Central University, Taoyuan City 32001, Taiwan (R.O.C), chia@csie.ncu.edu.tw

by introducing m.Site (2012) to allow site administrators to visually select web objects and assign attributes for adapting existing websites to mobile paradigm. However, Highlight and m.Site require administrators with front-end and back-end Web programming knowledge to modify or add features.

On the other hand, there are also tools for users with no programming background to create RWD websites. For example, WordPress, Wix and Weebly, etc. could generate HTML code for mobile webpages based on user specifications. Therefore, there would not be connection to existing websites. If users rely on the original desktop website for list item management, they may need to either manage two websites, or abandon the original website by creating responsible webpage for backend management. However, if we can reuse data from the existing websites and build linkages to the generated RWD website, the management efforts could be greatly reduced via automatic synchronization. As a result, we propose the DeXaR project to build a mobile website that allows users to focus on creating mobile websites instead of modifying the original website.

3. System Design

In order to quickly and easily create an RWD website from an existing website, we have developed two subsystems for responsible website creation and data ETL API creation. In addition to these two subsystems, we also develop deep web data extraction solutions. Due to space limitation, we focus on mobile website creation in this paper and give only overview of web data ETL system.

3.1 Mobile Website Creation

The mobile website creation provides two functions for page creation/editing and request response. As shown in Fig. 1, the website editor offers GUI (graphical user interface) for page creation and editing and keeps the configuration for each website in NoSQL (Not only SQL) database MangoDB. Upon the request of a page, the request renderer module coordinates with the request combiner to generate HTML and Javascript codes for browsing in mobile devices. Similarly, the page editor renderer coordinates with editor combiner to modify the GUI for page preview.

(1) Website Editor

Similar to commercial mobile website creator, the system also provide predefined templates, which are composed of web components. Users can select from predefined templates to create an RWD webpage. As shown in Fig. 1, the website editor manages mobile websites created by all users. Each website is maintained by a website manager class, which manages all web pages in a website. The settings for each website or webpage are stored in the Website Config Database. When adding or editing a web page, the page editor renderer calls editor combiner to produce the corresponding web code for page preview.

(2) Page Production

There are two cases that we need to generate HTML and Javascript codes for the created pages: one for page preview and one for official page request. Therefore, we have editor combiner and official combiner for respective webpage production. When a user browses a mobile webpage on smart handheld devices, the corresponding configuration is retrieved from the Website Config Database by the page request renderer which calls request combiner to generate the corresponding Web codes.

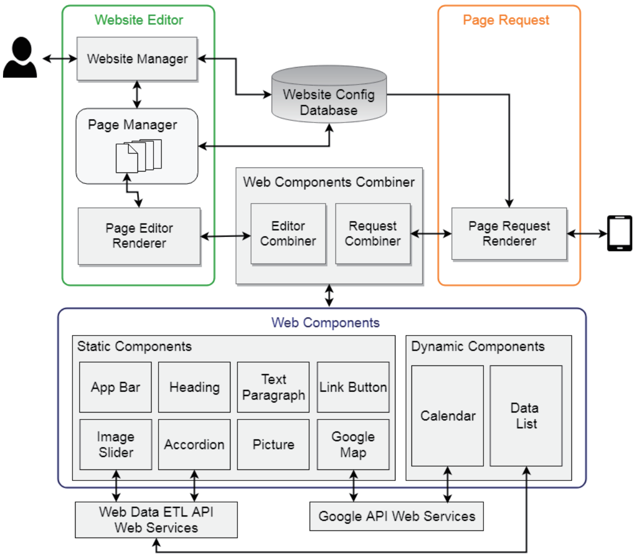


Fig. 1 Mobile Website Creator Architecture

(3) Static and Dynamic Web Components

To enrich the generated webpages, we have implemented a variety of web components including banner bar, heading, text paragraph, picture, link button, image slider, accordion, data list, calendar, maps, etc. Each web component includes an editor and a generator. The former is used in website editor for data setting, while the latter is used in web component combiner for code generation. Some web components use direct data from website Config database to generate static HTML code, while some use indirect access to extract data from data API dynamically.

(4) Design Methodology

To design a web-based development tool, we must consider the fluency of the user interface and the reusability of the self-made web component and the maintainability of the entire program. Therefore, we have adopted front-end engineering solutions React[1] to support single page application. React makes good use of JavaScript to manage the rendering and DOM object operation by encapsulating the elements of a webpage into components. Thus, whenever there is a change in the data on the page, the corresponding component will automatically reflect the change of the screen, which is very helpful for dealing with complex webpage interaction.

However, this kind of client-side rendering starts slowly since the main HTML content needs to be generated in the first request. To solve the above problem, we also use Node.js[2] since it can run JavaScript on the server side. By running Client-Side Rendering application on the server side, we can produce a complete page earlier. This not only improves the browsing experience on page loading, but also helps the search engine optimization (SEO) ranking of the mobile webpages.

[1] React, https://reactjs.org/

[2] Node.js, https://nodejs.org/

3.2 Web Data ETL System

As mentioned before, Web data ETL system makes use of web scraping and data extraction techniques for data API generation. Web scraping involves fetching or crawling a page based on HTTP requests and scrape data for extraction based on HTML parsers such as Tidy, CyberNeko, BeautifulSoup, etc.

In the DeXaR project, we adopt unsupervised web data extraction systems including single-page record set extraction system MDR [Liu2003] and multiple-pages detail data extraction algorithm AFIS [Yuliana2018]. Through intuitive and easy-to-use graphical interface, users can specify how to crawl, extract, and output data results (e.g., data API endpoint or static export) automatically without writing a program. Note that there is no need to label data for training (as for supervised approaches). Instead, users only need to select data columns in the spreadsheet as shown in Fig. 6. The generated Web data API can be used in many applications such as synchronizing an existing website with its RWD version through dynamic data API services.

4. Demonstration[1]

In this section, we illustrate how to create a mobile website from an existing website. We will also show how data APIs generated by Web Data ETL System is used in creating mobile webpages for Hopkins West Junior High (HWJHS) school. This is a typical case with no mobile website and IT support. The homepage contains several data regions including banner, navigation bar, image slider, news list, event calendar, etc. By clicking on news article, we obtain more news list as shown in Fig. 2.

To create responsible webpages, users can choose a predefined template such as dashboard, news list, maps, calendar, etc. as shown in Fig. 3. Through website editor, users are free to add or delete web components as show in Fig. 4 where the left navigation bar shows the web components added in the current page. For static web components, the data can be copied-pasted from existing webpage or manually input. For dynamic web combiners, users can specify the extractor name and the corresponding data columns to be displayed (see Fig. 5), where the extractor is generated via web data ETL system. As shown in Fig. 6, we create a data API with multiple-page input from HWJH news pages (Fig. 2) to obtain the tabular output. Fig. 7 shows the generated RWD webpages.

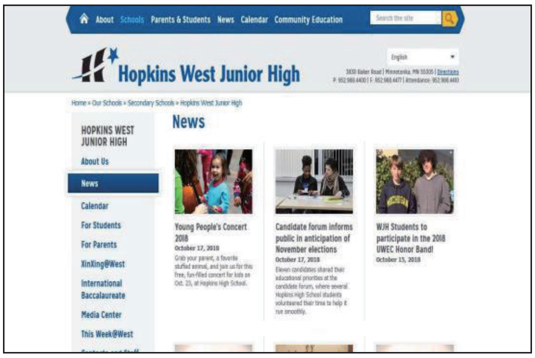


Fig. 2 The News Page of Hopkins West Junior High (HWJH)

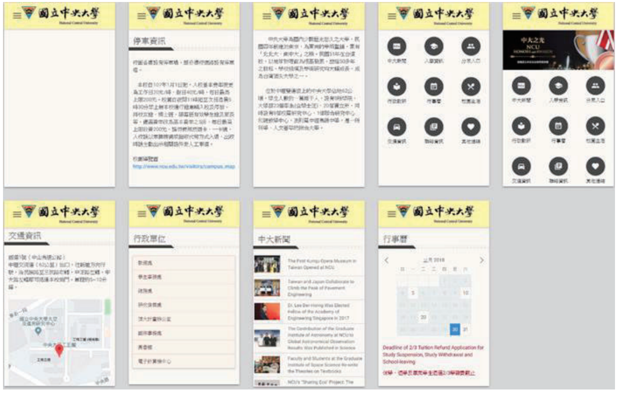
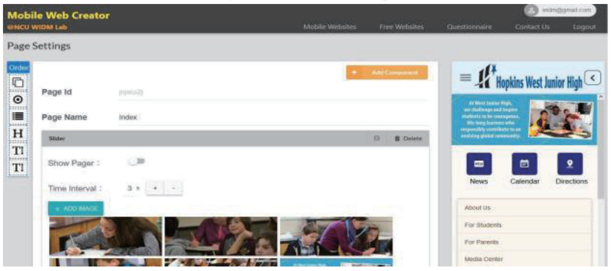


Fig. 3 Predefined Template



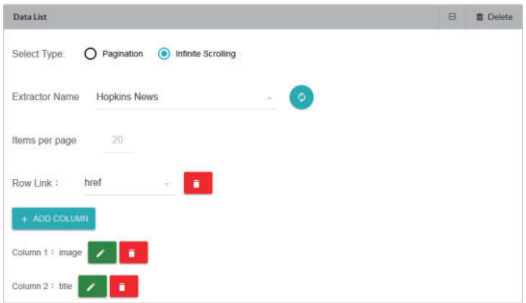
Page Id

Page Name index

Show Pager :

Time Interval : 3 s + -

Fig. 4 Mobile webpage editing illustration (Page Editor)



Data List ⊟ Delete

Select Type: ○ Pagination ● Infinite Scrolling

Extractor Name Hopkins News

Items per page 20

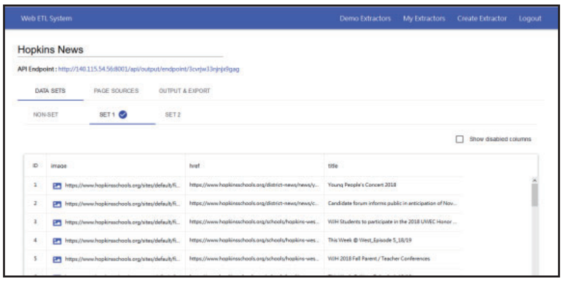
Row Link : href

+ ADD COLUMN

Column 1 : image

Column 2 : title

Fig. 5 Data List setting screen preview screen



Hopkins News

API Endpoint : http://140.115.54.56:8001/api/output/endpoint/3cvrjw33njnjx9gag

□ Show disabled columns

ID	image	href	title
1	https://www.hopkinsschools.org/sites/default/fi...	https://www.hopkinsschools.org/district-news/news/y...	Young People's Concert 2018
2	https://www.hopkinsschools.org/sites/default/fi...	https://www.hopkinsschools.org/district-news/news/c...	Candidate forum informs public in anticipation of Nov...
3	https://www.hopkinsschools.org/sites/default/fi...	https://www.hopkinsschools.org/schools/hopkins-wes...	WJH Students to participate in the 2018 UWEC Honor ...
4	https://www.hopkinsschools.org/sites/default/fi...	https://www.hopkinsschools.org/schools/hopkins-wes...	This Week @ West_Episode 5_18/19
5	https://www.hopkinsschools.org/sites/default/fi...	https://www.hopkinsschools.org/schools/hopkins-wes...	WJH 2018 Fall Parent / Teacher Conferences

Fig. 6 Data extraction result from Web Data ETL System

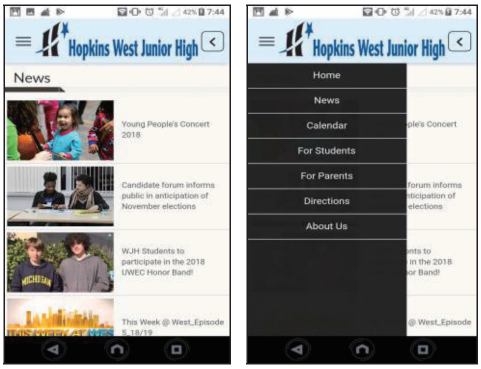

42% 7:44
Hopkins West Junior High
News
Young People's Concert 2018
Candidate forum informs public in anticipation of November elections
WJH Students to participate in the 2018 UWEC Honor Band!
This Week @ West_Episode 5_18/19
42% 7:44
Hopkins West Junior High
Home
News
Calendar
For Students
For Parents
Directions
About Us

Fig. 7 A mobile webpage created based on the news page from HWJH

[1] https://sites.google.com/site/nculab/project/dexar

5. Evaluation and Discussion

To evaluate the effectiveness of the proposed approaches, we run open demonstration several times and invite students to use the system to build a mobile website for any existing website. Then, a questionnaire of 7 questions (Table 1) is given to the test users. A total of 35 subjects replied the questionnaire. The scores for each question are given by 1~5 points, where one denotes very bad, very disagreeable or very difficult, and 5 denotes very good, very agree or very easy.

Table 1 Question list for the proposed questionnaire

	Questions
Q1	The interface is simple and easy to understand
Q2	The interface is easy to operate with high fluency
Q3	I don't need expertise to use the system
Q4	How easy is the tool to create a mobile website?
Q5	The completeness of the system in terms of functions and web components
Q6	I am interested in mobile website creation systems after using this tool
Q7	The quality and practicality of the established websites by this tool

We divide the users into two groups: with or without programming experience. Fig. 8 shows the experimental results. Students without web programming background gives higher score to the mobile website creation system than students with web programming background. Overall, most users can easily create a simple mobile website even without having any programming expertise. The interface is simple and easy to operate with fluent operational flow design.

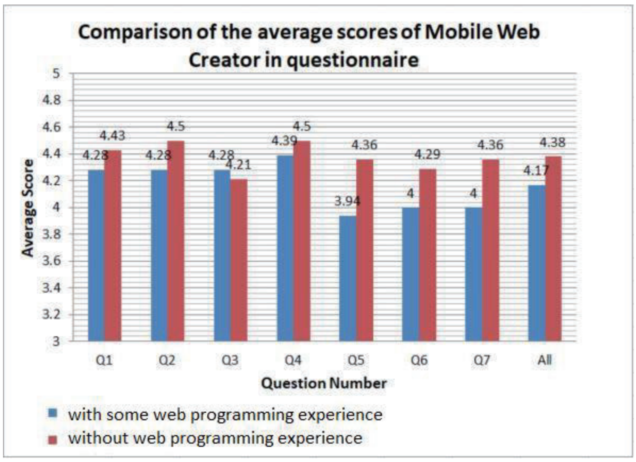


Fig. 8 Questionnaire scores for the mobile website creator system

While the experimental results seem to be great, most students only use static web components to create the mobile websites. Thus, it still lacks the evidence how web data ETL system can enhance the creation of mobile websites. Therefore, we plan to have the second run experiment to see whether web data ETL system could speed the mobile webpage creation in the future.

References

[Goto2016] K. Goto, & M. Toyama. Mobile web application generation features for SuperSQL. In Proceedings of the 20th International Database Engineering & Applications Symposium, 308-315.

[Koehl2012] A. Koehl, & H. Wang. m.Site: efficient content adaptation for mobile devices. In Proceedings of the 13th International Middleware Conference, 41-60. Springer-Verlag New York, Inc.

[Mohan1999] R. Mohan, J. R. Smith, C.S. Li, Adapting multimedia internet content for universal access. IEEE Transactions on Multimedia, 1(1), 104-114.

[Nichols2008] J. Nichols, Z. Hua, & J. Barton. Highlight: a system for creating and deploying mobile web applications. In Proceedings of the 21st annual ACM symposium on User interface software and technology, 249-258.

[Yuliana2018] O. Y. Yuliana and C.-H. Chang, A novel alignment algorithm for effective web data extraction from singleton-item pages, Applied Intelligence, 1–16.

[Toyama1998] M. Toyama. SuperSQL: an extended SQL for database publishing and presentation. In ACM SIGMOD Record, 27(2), 584-586.

[Xiao2009] I. Xiao, X. Luo, Q. Hong, D. Fu, H. Xie, & W.Y. Ma, W. Y. Browsing on small displays by transforming Web pages into hierarchically structured subpages. ACM Transactions on the Web (TWEB), 3(1), 4.