Abstract — Recent progress in the Web technologies lead to rapid expansion of Web applications, which started to look and feel like desktop applications. As the border between desktop and Web applications is blurred, a new concept emerged – Web operative systems. These systems represent software suite run in browser, hosting a number of services and applications. While general purpose and usability of this concept is still questionable, we can identify certain areas where this concept proved useful. We used this concept to build the virtual laboratory – specialized Web operative system consisted of various tools and services providing an educational support.

Keywords— Web applications, User interface, e-Learning

I. INTRODUCTION

Web development is already the most popular form of application development. Also, with the introduction of modern applications like Google Maps, Gmail, Writely, Flickr, etc., people started to use Web applications in a new way, for things previously done exclusively by desktop applications. We can see tendency that every application should be a Web application. By the term Web application it is meant every application which is run in a browser. Lately, there has been ongoing debate about which types of applications are preferred – desktop or Web. While each one has its advantages and disadvantages, it is important to understand the role and purpose of the Web applications. If we want to develop an online office application we already have tools to do it. But what about online video editing? Will it ever be possible to do it? If it is be possible, will it ever be considered as an efficient way of using resources?

The Internet has changed how we access and use information. With a computer and a high-speed connection, no matter where we are, our world travels with us. Of course, the reality is sometimes not so perfect, as we can’t have Internet access all the time, like when traveling through some remote locations or due to technical problems. Nevertheless, people are becoming more and more dependent on Web, which can be considered as a medium for a growing number of Web applications. Gmail and other webmail accounts completely changed the way people read their email. As users’ demands rise, need to further improve Web application’s presentation possibilities is also apparent.

Along with the demands, it is also of big importance to analyze what is the purpose of Web applications, or what area of problems they efficiently solve. Answering that question will give us the answer what will be the future of Web applications along with required technologies.

In this paper, we will examine purpose of Web operative systems and its application in education. Differences between desktop and web applications will be presented in the second section. Third section describes Web operative systems, while fourth section describes virtual laboratory, specialized Web operative system.

II. DESKTOP VS. WEB APPLICATIONS

Main purpose of Web is medium for collaboration, communication and making information available. It does not matter if it is being accessed through browsers or other applications. Web can be used by desktop applications as well, but then they can go back to the local computer for intensive processing. This is one of the main advantages of desktop applications over Web applications. Now days, we can see blending of the desktop and Web applications, each one borrowing features of the other. So, Web applications look more similar to desktop applications, and desktop applications are now “webified”, improved in a way they can benefit from the connectivity and mobility of web-based data. In this section we will analyze which approach in application development is better [2].

Clear victory goes to Web applications when we compare its delivery and compatibility. Since application is downloaded every time it is used, there is no need for installation on local computer and no access privileges are needed. Furthermore, we don’t have to worry about application updates. Compatibility is a big advantage since we can theoretically run Web applications on every computer with the browser. However, the value of compatibility is partial since it is based on the development of open standards. Even now we see that Web application’s interaction with the users is slightly different and not bug-free on different browsers and different platforms.

One of the problems in using Web applications is not using the ever-increasing performance and efficiency of the desktop processors. It can be considered as waste of the resources. With everything on the Web, why do we need faster desktops? Why don’t we go back to the older desktop operating systems and older hardware and use the Web? Apparently, not all problems can be solved by Web applications. And even if they could, why would we
ignore resources of the local PC and heavily depend on network accessibility and bandwidth.

Potentially big disadvantage of the Web applications is in their very nature – network dependency. We must be connected to Internet in order to use them. In some cases, this is normal, like when chatting or checking email. But what if we want to read already received emails? We can’t do it offline using Web applications. Or worse, if we happen to use some online calendar and we currently don’t have network access, we won’t be able to see our schedule. Besides network accessibility there is one more problem – server load and bandwidth. If server is currently under heavy load, there is a chance we can’t use Web applications on that server. This scenario is already happening.

Data storage is another question to consider. Web applications hosted on remote servers can’t use local databases. Also, security and privacy issue will likely to be against company policy. And what about exit strategies for the companies providing online services? What will happen if some company with a huge number of users decides that providing services is not profitable anymore? On the other hand, data centralization offers better protection against external intrusions (e.g. viruses, hackers) and better backup policy. Also, it’s really convenient for ordinary users that they can access personal data (emails, documents etc.) from everywhere in the world with only network connection needed.

There are many examples of desktop applications that benefit from the connectivity and mobility of web-based data: Word 2007 (allows blogging from Open API), Windows Live Writer (a powerful replacement for web-based blogging), iTunes (integrated with its online music store), and many other desktop applications which use Web, like offline search engines, multimedia organizers, data manipulation, etc.

As the Web becomes increasingly interconnected and applications continue to blur the distinction between the desktop and web, we should expect to see more applications that allow Web/desktop synchronization. This will happen due to the increasing development of web services that enable applications to work equally well across web and desktop clients.

Webified, internet deployable desktop applications - that can reliably store data, serve it robustly, and interact with both remote and local databases will become more and more popular. This connected model will ensure that applications will function in both online and offline states, providing a seamless, uninterrupted experience.

Nevertheless, desktop and web are just small outposts in a much larger world of information creation, management, collaboration, and presentation. What ultimately matters is productivity, scalability and speed.

Desktop applications will always have its range of problems they solve better than Web applications. On the other hand, Web applications will continue to “take over territory” of desktop applications [7][8]. Most obvious example is webmail expansion or new Microsoft’s project – Microsoft Office Live where user will use office-like Web applications instead of conventional, desktop alternatives [1].

### III. WEB OPERATIVE SYSTEMS

Web Operative System (or just WebOS or Webtop) provides basic operating system services needed to build applications that are geographically distributed, highly available and dynamically reconfiguring [3]. It is more a suite consisted of various applications with the look and feel of the desktop work area.

Major advantage of WebOS is in its accessibility and delivery: users can access their personalized desktop from everywhere and immediately use applications.

This concept adds one layer of abstraction between software and hardware. It doesn’t even matter which OS is running the hardware – for a WebOS user it is transparent (Figure 1).

![Figure 1 WebOS hierarchy](image)

WebOS is relatively new and interesting concept, but for many people with arguably questionable value. It can be fun, exciting, entertaining and even convenient for some users – but being as efficient, flexible and productive as a desktop is practically impossible.

Major downsides of WebOS are inherited from their nature of being Web applications: network dependency and high bandwidth, sensitive data storage, performance issues, inability to operate peripheral devices and constantly changing array of technologies, often not backward compatible.

Following is a list of currently available popular Webtops:


**DesktopTwo** [5] provides email, address book, file storage and sharing, IM, blog, music player and a website editor in a nicely organized user-interface. It requires Flash, Acrobat Reader, and popup windows to function correctly.

**YouOS** [6] has perhaps the most recognition of all the WebOS products. The YouOS developers describe their product as “a liberation of software from hardware”. YouOS wants the OS to be no longer a user's primary
concern - it's your data and your apps that you only need to concern yourself with.

IV. VIRTUAL LABORATORY

Idea of having an online virtual laboratory included into our curriculum has been conceived on Department of Computer Science, Faculty of Electronic Engineering of Niš [9][10]. Our primary goal was to develop an application which will allow students to learn, practice and even evaluate their knowledge. Since this application must be consisted of several applications, we realized that desktop-like environment is the most appropriate. It became apparent that WebOS is ideal solution for this problem.

![Virtual Laboratory screenshot](image)

Our system is intended to support education in the fields of automatic controls, logic design, image processing, programming and Internet technologies. It is capable to handle multiple windows at the same time (like in Windows, see Figure 2), provides high level of personalization for each user, it is customizable and it uses asynchronous loading. This loading approach is currently very popular because of AJAX, which is based on this concept. Aforementioned approach can improve loading times and cut the redundancy.

Main benefits of the virtual laboratory are high level of interaction with the student and easier learning and practicing due to access to multimedia content. Also, student can use laboratory from remote computer and his results are stored on the Web, enabling his progress to be monitored.

Student is able to open several applications on his virtual desktop at the same time. For example, student can open tutorial, practical online exercise and Web search, each in separate windows. He can personalize look and feel of the virtual desktop – arrangement of icons, main menu and startup applications. System is designed to have built-in calendar and scheduler. That way student can quickly determine important dates and notes. It is noticeable that organizer services are also available in most current Webtops.

Development of this system is possible because of the object oriented framework for Web application’s client side [11]. Framework has support for rapid development of Web application’s client side. Main benefits of the framework are:

- Object oriented approach
- Separation of control, content and presentation
- Elimination of redundancy
- Decreased development time and costs
- Better teamwork organization
- Great flexibility and robustness
- CASE tools support
- Compatibility with W3C standards
- Multilanguage controls
- Asynchronous loading

Architecture of the user interface (Web page) along with various page elements can be easily created. More specifically, virtual laboratory is consisted of the main page (desktop work area) and other windows which can present other Web pages or services (Figure 3).

![Communication between framework and Webtop](image)

Opened windows within laboratory Webtop can communicate with each other enabling high level of interactivity with the user.

Framework in consisted of several subsystems (Figure 4). Each subsystem is responsible for specific domain of actions.

Core is essential part of the framework, integrating all other subsystems into one compact system.

Control Subsystem is bridge between the framework (more specifically Core) and all other Web pages. This subsystem contains business logic which manage all resources in the application (windows, controls, display, etc.)
Display Subsystem manages user interface architecture and all containing graphic. It makes Webtop to be skinnable, providing themes support. Themes can be changes in the runtime.

Navigation Subsystem is responsible for all navigation menus on the page. Since it is has Model-View-Controller architecture, menu presentation and its contents are separate. This separation provides runtime changing of both menu data structure and its presentation. For example, we can add or delete menu items, and also change its presentation from standard horizontal menu to popup menu.

Windows Management Subsystem provides features of the desktop – opening, dragging, overlapping and closing windows. Each window’s outlook can be changed in the runtime, and several different window outlooks can be present at the same time.

Document Management System manages all Web documents loaded within the Web page. It also provides advanced user controls features: multilanguage support, replaceable set of icons and communication possibilities with other controls even in other documents.

All of above mentioned subsystems are building blocks for the virtual laboratory Webtop, providing advanced user interface with the high level of user interaction.

V. CONCLUSION

It can be concluded that WebOS concept can indeed be practical, but for limited and specialized number of scenarios (e.g. office applications). Ongoing tendency that Web applications take over all problems previously solved by desktop applications is not real, and time will tell that both approaches will have its exclusive set of problems they solve. In the same way, webtops will find their purpose, but won’t be able to completely replace real desktop operative systems. Online virtual laboratory is one scenario which can make use of Webtops. Further development of virtual laboratory will include integrating more tools, like forum, support for mobile phones, etc.

LITERATURE