

## **DecSoft HTML Compiler**

### **History**

#### **2024.5 (04/26/2023)**

\* Add the new `dhc.system.macAddresses()` external JavaScript function. This function can retrieve the network devices MAC addresses of the PC in which our app is running. The function is intended mainly to use these MAC addresses in order to get a more or less unique identifier of the PC in which our app is running, so we can use it, for example, to license our app using that possible unique ID.

\* Add the new `MacAddresses` sample app, which show how to use the new `dhc.system.macAddresses()` external JavaScript function in our apps. Of course, this new function is also described in the product help.

#### **2024.4 (03/28/2023)**

\* Update the Delphi library that we use with the OpenAI assistant to the latest version. We need to do some internal changes in order to use the latest version of the Delphi OpenAI library, and, move on to use the GPT4 model instead of the "GPT4-preview" one for the OpenAI assistant chat capability.

#### **2024.3 (03/06/2024)**

\* Update the Font Awesome JavaScript library to the latest 6.5.1 version. We use the Font Awesome JavaScript library in the DecSoft HTML Compiler About page, the product help and the Welcome page, as well in the product's ReadMe file.

#### **2024.2 (02/20/2024)**

\* Update the Bootstrap CSS framework (CSS and JavaScript) to the latest 5.3.3 version for the program's welcome page, program's help, program's about and program's "read me" file. This version of Bootstrap CSS includes a lot of changes, fixes and enhancements.

#### **2024.1 (01/01/2024)**

\* This is a major release of this product and require a new license (or upgrade license) if you purchase it before 01-01-2023. You can purchase a license from our store, and, an upgrade license (if you needed) from your DecSoft customer area, with a 50% off discount. Also from

your DecSoft customer area you can get the outdated releases of the product.

#### **2023.23 (12/05/2023)**

\* Replace the "text-davinci-003" model for code generation in the Open AI assistant with the recommended model "gpt-3.5-turbo-instruct". The "text-davinci-003" model has been marked as deprecated and become not available in January 2024.

#### **2023.22 (11/13/2023)**

\* Enhance the Open AI assistant which now use the new Chat-GPT-4 model instead of the Chat-GPT-3.5 model, and, the new DALL-E 3 model instead of DALL-E 2 model. The new models offer better results, for example, the Chat-GPT-4 model training data is newest and the DALL-E 3 model (for images generation) produces better images.

#### **2023.21 (09/14/2023)**

\* Update the Bootstrap CSS framework (CSS and JavaScript) to the latest 5.3.2 version for the program's welcome page, program's help, program's about and program's "read me" file. This version of Bootstrap CSS includes a lot of changes, fixes and enhancements.

#### **2023.20 (09/13/2023)**

\* Update the Delphi library that we use with the OpenAI assistant to the latest version. In the latest version of the referred library, "[...] the API has been fully updated in accordance with the official [OpenAI] version as of September 13, 2023". The library also has other changes, fixes and enhancements, so our OpenAI assistant.

#### **2023.19 (08/29/2023)**

\* Update the jQuery JavaScript library to the latest version (3.7.1) in the program about page, program welcome page, program help pages, various sample apps and the program "readme" file. This new version of jQuery contains various fix, changes and enhancements.

#### **2023.18 (08/28/2023)**

\* If we start the program without an Open AI API key defined in the program options, and then we establish an Open AI API key from the program options, and then we try to use the

integrated Open AI assistant, we will experience an error, which basically means that we did not establish an Open AI API key. Once the program restarts, the behavior is as expected, but this release fixes the referred-to wrong behavior, so we can start to use the Open AI Assistant once we set an Open AI API key from the program options, that is, without the need to restart the program at all.

#### **2023.17 (08/26/2023)**

- \* Improve the Open AI assistant by providing a multiline control for the prompt. This allows us to prepare a more convenient prompt, especially in some circumstances, like when we want to translate certain more or less large texts that can include multiple lines. Make other minor improvements, like trimming the prompt before sending it to Open AI.

#### **2023.16 (08/15/2023)**

- \* Fix an exception error if we try to search for new product releases when there is no internet connection. Now in this specific case, we get the right and expected error message (no host found; check the internet) instead of a weird exception that does not offer clear information about what is happening.

#### **2023.15 (07/26/2023)**

- \* Update the Bootstrap CSS framework (CSS and JavaScript) to the latest 5.3.1 version for the program's welcome page, program's help and program's about. This version of Bootstrap CSS includes a lot of changes, fixes and enhancements.

#### **2023.14 (07/05/2023)**

- \* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits. These installers are used by DecSoft HTML Compiler to offer the preview of your HTML apps inside the IDE, and, also, optionally (it's completely up to you as you probably already know) to be included by your app compiled executables.

#### **2023.13 (06/05/2023)**

- \* This release is a major version of the product. If you purchase DecSoft HTML Compiler at or before 2022-06-01, you can purchase an upgrade license (with a 50% off discount) from

your DecSoft Customer Area at our website, in order to register this release and the future minor releases for another year. Your current license does not expire, but it's now outdated: you can download the right outdated release also from your DecSoft Customer Area at our website. If you purchase DecSoft HTML Compiler after 2022-06-01, you can also log in to your DecSoft Customer Area in order to grab your updated license serial number. Please, don't hesitate to ask us if you have any questions. Thanks in advance for your support and trust in our work!

#### **2023.12 (05/30/2023)**

- \* Update the Bootstrap CSS framework (CSS and JavaScript) to the latest 5.3.0 version for the program's welcome page, program's help and program's about. This version of Bootstrap CSS includes a lot of changes, fixes and enhancements.

#### **2023.11 (05/24/2023)**

- \* We discover a rare bug which is caused when try to load the app icon (not all, only some specific app icon files) in the app options dialog. This rare bug can cause an unexpected behaviour in the program, however, the "wrong icon" may can be used in the app executables, so, we decide to remove the "icon preview" from the app options dialog, avoiding this rare error and therefore the possible program unexpected behaviour.

#### **2023.10 (05/12/2023)**

- \* Update the jQuery JavaScript library to the latest 3.7.0 version for the program's welcome page, program's help, program's about and various sample apps. This version of the jQuery library includes a lot of changes, fixes and enhancements.

- \* Update the Font Awesome JavaScript library to the latest 6.4.0 version for the program's welcome page, program's help and program's about. This version of the Font Awesome JavaScript library includes a lot of changes, fixes and enhancements.

- \* Update the Bootstrap CSS framework (CSS and JavaScript) to the latest 5.3.0 (alpha3) version for the program's welcome page, program's help and program's about. This version of Bootstrap CSS includes a lot of changes, fixes and enhancements.

#### **2023.9 (04/21/2023)**

- \* As you probably already know, DecSoft HTML Compiler never extract your compiled app files into the user computer. The compiled app files stay inside the compiled executables

and run from the executables. This release improve the security of the compiled app inside the compiled executables. This do not means that before this release the compiled app can be easily accessed (not at all), but, we are now using an even more advanced method in order to increase the security of your compiled app files.

#### **2023.8 (03/29/2023)**

\* Improve the Open AI assistant by changing some texts of the interface, but, mainly, adding a new tab which allows us to generate images from Open AI. So we can now generate source code, maintain a chat conversation and generating images using Open AI directly from your favorite software, DecSoft HTML Compiler! :-)

#### **2023.7 (03/28/2023)**

\* This release of the product incorporates an Open AI assistant (you can see it in the Menu -> Tools -> Open AI assistant) in order to be use the Open AI API directly from the IDE. You must provide an Open AI API key in order to properly work. The Open AI assistant can be used to ask for some code solution, like a "JavaScript function to validate EMail addresses", and, also to maintain a chat with Open AI, in order to get some other kind of solutions, like translated texts and many more.

#### **2023.6 (03/25/2023)**

\* Update the Bootstrap CSS framework (CSS and JavaScript) to the latest 5.3.0 (alpha2) version for the program's welcome page, program's help and program's about. This version of Bootstrap CSS includes a lot of changes, fixes and enhancements.

#### **2023.5 (03/09/2023)**

\* Update the jQuery JavaScript library to the latest version. This means the jQuery JavaScript library used in the product "read me" file, the product About page, the product Welcome page, the product help and some sample apps who uses it.

#### **2023.4 (02/19/2023)**

\* Fixed a bug in the XML library used to work with the project XML files which causes that we can't properly retrieve previously saved data (in CDATA nodes) containing two closing square brackets without spaces: "]]". Sometimes it's possible to "simply" place some space

between the brackets, but, the better way to solve this is the way in which this release act: using a fixed XML library which now have no problems to deal with the referred two closing square brackets. This fixed bug is more important in our DecSoft App Builder product, however, since DecSoft HTML Compiler uses the same XML library to work with the project XML files.

### **2023.3 (02/09/2023)**

- \* Enhance the product help in order to describe the functionality of the recently added `dhc.getBorder()` and `dhc.setBorder()` JavaScript external functions.
- \* Enhance the Window sample app in order to incorporate the recently added `dhc.getBorder()` and `dhc.setBorder()` JavaScript external functions.
- \* Add the new JavaScript external function `dhc.setBorder()` which give us the ability to establish at runtime the app window border that we can also establish at designtime.
- \* Add the new JavaScript external function `dhc.getBorder()` which give us the ability to know at runtime the app window border established in the app at designtime.

### **2023.2 (01/28/2023)**

- \* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.
- \* By some kind of "tradition" (probably from the arrival of XHTML long time ago) we start to use trailing slashes in void HTML elements, like images, breaks, horizontal rules, inputs, etc. That trailing slashes in such elements are not certainly required (talking about HTML), and, in fact, if we validate the HTML markup using the W3C validator, for example, we get an "info" reminder us that that trailing slashes for void elements are not really required in HTML documents. Well. We remove from our mind that "tradition" and therefore remove the referred trailing slashes from the DecSoft website and all our products. In particular, in this product we remove the referred trailing slashes from the HTML of the welcome page, about page, help pages and app samples.

### **2023.1 (01/01/2023)**

- \* After 26 releases in 2022, here is the first release of the product in 2023. This is a major release of the product, means if you purchase your license before 01-01-2022, you must purchase an update license (with a 50% off discount) or continue using your outdated license with the last outdated release of the product, that you can download from your

DecSoft customer area. Wish you the best for you and your family in this new year!

#### **2022.26 (12/25/2022)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.3.0 version for the program's welcome page, program's help and program's about. This version of Bootstrap CSS introducing the support for the "dark mode", and, we start to use it in the program's welcome page, program's help and program's about. Additionally, we enhance the help code editors (wich show samples of the external JavaScript functions) when the program is running in dark mode too.

#### **2022.25 (12/23/2022)**

\* Add the new `dhc.dialogs.selFolder()` external JavaScript function, document it, and, enhance the Files sample app with a sample of this function call. This function is intended to let the user to choose a folder path from the file system.

\* Add the new Construct3 sample app. This sample app show us how easy is to compile with DecSoft HTML Compiler a game created with Construct 3. In fact, it's as easy as any other HTML app, in this case a HTML game. Thanks to DecSoft HTML Compiler we can compile our Construct 3 games into standalone executables for Windows 32 and 64 bits, without expose the game source code.

#### **2022.24 (12/21/2022)**

\* Update the jQuery JavaScript library to the latest version. This means the jQuery JavaScript library used in the product "read me" file, the product About page, the product Welcome page and the product help.

#### **2022.23 (12/16/2022)**

\* Update the jQuery JavaScript library to the latest version. This means the jQuery JavaScript library used in the product "read me" file, the product About page, the product Welcome page and the product help.

#### **2022.22 (12/12/2022)**

\* Due to a stupid bandwidth starvation attack to our server, we are obligated to perform some changes in how the product updater works. This means two things: this release made

the right changes in order to get the updater working as expected after the referred changes, and, unfortunately, the previous release of the product cannot be updated using the product updater. We apologize for the possible inconveniences. If you are trying to update the product to this release using the product updater, please, just download the latest release of the product from our website.

#### **2022.21 (11/22/2022)**

- \* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.2.3 version for the program's welcome page, program's help and program's about.

#### **2022.20 (10/06/2022)**

- \* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.2.2 version for the program's welcome page, program's help and program's about.

#### **2022.19 (09/18/2022)**

- \* Add the WinSize sample app, which show us how to use the recently added `dhc.window.getSize()` and `dhc.window.setSize()`, in order to save the position and size of the app window when it's closed, and, set the saved position and size when the app is started again, in this case saving the information in a JSON file.

- \* Add the new `dhc.window.setSize()` Javascript external function. With this function we can establish the position (top and left) and the size (width and height) of our app window in runtime. We can configure the app position and size at designtime already, and, with this function, we can set it now also at runtime.

- \* Add the new `dhc.window.getSize()` Javascript external function. We can use this function to retrieve the current app window position (top and left) and size (width and height) at runtime. We can use this information, for example, to use it when the app is closed, and, therefore, load it when the app is started.

#### **2022.18 (09/12/2022)**

- \* Add the new TrayMenu, TrayIcon and TrayIcon2 sample apps. These apps shows how we can use the recently added Javascript external functions and events related with the app tray icon and the app tray icon menu. The product help has been also updated in order to incorporate these new stuff related with the app tray icon and the app tray icon menu.



\* Add the new `dhc.events.onTrayMenuItemClick()` Javascript external function. This function / event is called when the user click on an item of the app tray icon menu and receives the item ID which has been clicked as an argument. Take a look at the product help and also at the TrayMenu sample app to see this function / event in action.

\* Add the new `dhc.events.onTrayIconClick()` Javascript external function. This function / event is called when the user click on the app tray icon. Take a look at the product help and also at the TrayMenu, TrayIcon and TrayIcon2 sample apps to see this function / event in action, and, also to look at the related tray icon new external functions.

\* Add the new `dhc.trayMenu.hideIcon()` Javascript external function. With this function we can hide the app tray icon if we previously show it by using the also new `dhc.trayMenu.showIcon()` Javascript external function. Take a look at the TrayIcon and TrayIcon2 sample apps to see this in action.

\* Add the new `dhc.trayMenu.showIcon()` Javascript external function. This function can be called (at runtime) in order to show the app tray icon in the system tray. The behaviour of the `dhc.window.setState()` action has been changed in order to minimize the app into the system tray (hidden it from the task bar) if the tray icon is shown. This allows us, for example, to start the app running in the background / system tray. Take a look at the TrayIcon2 sample app to see this in action.

\* Add the new `dhc.trayMenu.updateItem()` Javascript external function. This function works in a similar way than the already existing `dhc.mainMenu.updateItem()` and update an specific item of the app tray icon menu. Take a look at the product help and also at the TrayMenu sample app to see this new external function in action.

\* Add the new `dhc.trayMenu.setItems()` Javascript external function. This function works in a similar way than the already existing `dhc.mainMenu.setItems()` and set the app tray icon menu items. Take a look at the product help and also at the TrayMenu sample app to see this new external function in action.

## **2022.17 (09/11/2022)**

\* This release of the product incorporates an internal change which we already incorporates in our DecSoft App Builder product. What we do in this DecSoft HTML Compiler release is to save the app project files in a more smart way, that is, by including in the project file only the app options which has been certainly changed, or, in other words, by not saving on the project file the app options which are not changed, which have their default values. Doing this what we get are smaller project files, but, also smarter files, since do not include options values which are not completely required, since still with their default values.

#### **2022.16 (09/10/2022)**

\* This is a maintenance release of the product with some minor internal changes, fixes and enhancements. Basically in a previous silent release we change the Webview arguments for the IDE Webview, in order to get something working in the About, Welcome and Help pages, however, we decide to remove that stuff, and, also we remember that our IDE Webview must work exactly with the right arguments than the default apps WebView, in order to allow us to see our apps running as expected in the IDE Webview too.

#### **2022.15 (09/09/2022)**

\* In order to get our apps properly working DecSoft HTML Compiler establish some flags / options for the Edge Runtime Webview in which our apps are running. This release of the product add a new app option which allows to modify or add new flags / options for the Edge Runtime Webview. As you can see in the product documentation, this app option must be carefully taken: they are intended only for advanced users who want to try or establish new flags / options or to modify the ones that DecSoft HTML Compiler establish by default. Thanks **Steven** for the idea!

#### **2022.14 (09/08/2022)**

\* Properly identify the MIME type of Javascript files. Previous to this release, we are not perfectly indicated (internally) the MIME type of the Javascript files of our app. This do not cause any trouble, except if we load the Javascript file with an script tag with the type="module" attribute. In this specific case, the MIME type will be "perfectly indicated" or the Javascript file cannot be properly loaded.

\* Avoid a problem which is caused if the CORS origin value of a Javascript file is "null", which is the default value for our Javascript files, since they are loaded using the "file" protocol instead of the HTTP(S) one. This release prepares the Edge Runtime WebView in order to allow that kind of CORS origin values, which means that some specific apps (like the Construct 3 software HTML5 exported games) work now as expected.

#### **2022.13 (09/07/2022)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.2.1 version for the program's welcome page, program's help and program's about.

\* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

## **2022.12 (09/02/2022)**

- \* Add the new Notifications sample app, which allows us to see the latest system notification stuff working. Basically we can see how to show various notifications and to react when these notifications are clicked by the user. Add also in the product help the right documentation for this new notifications related stuff.
- \* Add the new `dhc.events.onNotificationClick()` event, which is fired when a system notification which we previously shown has been clicked by the user. We can know what notification has been clicked because this event receive the name of the notification.
- \* Add the new `dhc.system.notification()` Javascript function, which allows us to show system notification to the user of our apps. We can establish the notification's name, title and body: in particular the name of the notification is important because we receive it in the new `dhc.events.onNotificationClick()` event when the notification is clicked by the user.

## **2022.11 (08/28/2022)**

- \* Update the jQuery Javascript library to the latest version in the product About, Welcome and Help pages. This is also a maintenance release of the product with other minor internal changes, fixes and enhancements.

## **2022.10 (08/21/2022)**

- \* The Video and Audio sample apps shows a couple of media controls, one of them with the "autoplay" option set to true. From some Edge Runtime WebView releases ago the "autoplay" start to work as expected: the user must previously interact with the page in order to allow the "autoplay" option of a media control. This release of the product set the appropriate flag for the Edge Runtime WebView to set the "autoplay" option working even without the interaction of the user, which is something that can be good for our apps: you can see now the Video and Audio sample apps working as expected again, for example.

## **2022.9 (08/19/2022)**

- \* Handle correctly the possible query string used in our app's files. For example, suppose we want to use (in some way) a file like `other.html?var=value`. Previous to this release, DecSoft HTML Compiler can't find that file in the executable, because certainly that file "as is" do not exists (what exists is "other.html"). This release fix this kind of problem so we can use query strings in our app files if we need that. Thanks **Dragan** for your error report!

## **2022.8 (08/07/2022)**

- \* Enhance the product help with the descriptions of the recently added `dhc.registry.*` external Javascript functions and add a new WinRegistry sample app, which show us how to use these recently added external functions.
- \* Add the new `dhc.registry.writeValue()` external Javascript function. This function allows to write string values in keys under the `HKEY_CURRENT_USER` Windows Registry root key.
- \* Add the new `dhc.registry.readValue()` external Javascript function. This function allows to read string values in keys under the `HKEY_CURRENT_USER` Windows Registry root key.
- \* Add the new `dhc.registry.valueExist()` external Javascript function. This function allows to check if certain string value exist in keys under the `HKEY_CURRENT_USER` Windows Registry root key.
- \* Add the new `dhc.registry.deleteValue()` external Javascript function. This function allows to delete a specific string value in keys under the `HKEY_CURRENT_USER` Windows Registry root key.
- \* Add the new `dhc.registry.keyExist()` external Javascript function. This function allows to check if certain key exist under the `HKEY_CURRENT_USER` Windows Registry root key.
- \* Add the new `dhc.registry.deleteKey()` external Javascript function. This function allows to delete a specific key under the `HKEY_CURRENT_USER` Windows Registry root key.

## **2022.7 (07/25/2022)**

- \* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.2.0 version for the program's welcome page, program's help and program's about.
- \* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.
- \* This is a major release of DecSoft HTML Compiler, means if you purchase the product before 07-25-2021, you must to purchase an upgrade license (with a 50% off discount), or download the outdated release to continue using your outdated license. If you purchase your license after the referred date, please, login into your DecSoft customer area to grab your new license serial number. You can purchase an upgrade license as well to download the outdated release from your DecSoft customer area in our website.

## **2022.6 (05/30/2022)**

\* Due to some unexpected changes in our hosting the updater tool of the product stop to working as always. This release do the appropriate changes in order to allow the updater tool to continue working as always, that is, with the ability to check if a new version exists, to download and show the history of the product and to download an existing new release.

\* Add the Geolocation sample app. This sample app show us how to get the app user geolocation latitude and longitude by using the appropriate Javascript API. This sample also show how we no need to ask for the right permissions from the user, since our apps already set the appropriate permission to get the geolocation information.

\* Change the behaviour when clicking the "Open a sample" button from the New App Index File dialog. Instead of trying to open an index.html file of the existing sample apps, what the button do now when clicked is to close the referred New App Index File dialog and open the sample apps folder: this is the same behaviour that we can see when use the main menu Samples item or the main toolbar Samples button.

## **2022.5 (04/22/2022)**

\* Fix a bug which cause that the menu items of the optional app's main menu are not properly drawn if the app uses an optional style different than the default one. Another minor changes and fixes has been also made for this release of the product.

## **2022.4 (03/04/2022)**

\* Add the new app's compilation option "Allow the user to change the zoom factor". This option can be check to true in order to allow the user to change the zoom factor of the app's executable browser. By default this option is not checked, mean the user can't change the zoom factor at runtime.

\* Add the new `dhc.window.getZoom()` external Javascript function. With this function you can get the currently established app's executable browser zoom factor. By default the zoom factor is established to "1", however, optionally it's possible to change and therefore you can use this Javascript function to determine the current zoom factor value.

\* Add the new `dhc.window.setZoom()` external Javascript function. With this function you can set the app's executable browser zoom factor. This function works independent app's compilation option which can allow the user to change the zoom factor at runtime. You can use this function if you want to change the zoom factor at runtime.

\* Add the new Zoom sample app. This sample app shows the usage of the new `dhc.window.getZoom()` and the `dhc.window.setZoom()` external Javascript functions added

to the product and available for your apps. You can look also at the app's Javascript option, since you can use this option in order to establish a zoom factor different than the default one, so the app starts with your desired zoom factor.

\* Document in the product help the new external Javascript functions and the new app's compilation option related with the app's executable browser zoom factor. You can read in the help about the new app's compilation option as well the new related external Javascript functions, even when you can also take a look at the new Zoom sample app.

### **2022.3 (03/02/2022)**

\* Add the new SQLite sample app. This new sample app show how we can use the SQL.js Javascript library in order to use SQLite databases from our apps. We can create databases from scratch, as well to load existing database files (even if we compile the database file in our apps executables) and save database files from our apps. The sample app is quite simple, but, at the same time, show how we can do the principal tasks to deal with SQLite databases: load a database file if exists, create a database table, save the database into a file and perform SQL queries to select and to insert new information into the database. Probably the two more important tasks that the sample show is how we can load a database from a file and how to save a database into a file. Take a look at the index.html file of the sample app for more information and details.

### **2022.2 (02/06/2022)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.1.3 version for the program's welcome page, program's help and program's about.

\* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

\* Add the new Clipboard sample app. This sample app show us how we can use the Javascript Clipboard API to directly copy and cut text to the system clipboard, as well how to paste text from the system clipboard to our apps.

### **2022.1 (01/01/2022)**

\* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

#### **2021.49 (11/30/2021)**

\* Our hosting silently starts to automatically redirect our HTTP calls to HTTPS and this cause problems while check for new releases of the product, for example, so in this release we add the appropriate stuff and enhance our internal procedures in order to use HTTPS by default, avoiding the problems. Thanks **Ade** for the report!

#### **2021.48 (11/20/2021)**

\* Add the new `dhc.app.getArgs()` external Javascript function. This function allows to retrieve the command line arguments passed to the app executable. Can be also used to retrieve the path of the app executable, since the argument number zero is always the app executable path.

\* Add the new Arguments sample app. This sample app show us how to use the new `dhc.app.getArgs()` function in order to retrieve the optional arguments passed to the app executable program. The sample list the arguments who is passed to the app executable. Take a look at the sample `ReadMe.txt` file for more information and details.

\* Add the new OpenWindow sample app. This sample app show us how we can load external URLs using a new app window, instead of a new default browser window. We use a intermediate HTML file (`redirect.html`) to load an external URL in a new app window. Take a look at the sample `ReadMe.txt` file for more information and details.

#### **2021.47 (10/24/2021)**

\* First release of the product compiled in Microsoft Windows 11. Fixed the detection of the Microsoft Windows version in order to be used internally at some places in the product. Set the rounded corners in the GUI forms when using the dark theme of the product GUI and do the same for the user compiled apps forms. Test the product at Microsoft Windows 11.

#### **2021.46 (10/06/2021)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.1.2 version for the program's welcome page, program's help and program's about.

\* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

**2021.45 (10/05/2021)**

\* By a mistake of mine, the previous release of DecSoft HTML Compiler lost the ability to apply the expected syntax highlighting in the sample source code snippets of the help: this release fix this mistake and therefore we can properly see the syntax highlighting applied in such sample source code snippets in the help.

**2021.44 (09/08/2021)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.1.1 version for the program's welcome page, program's help and program's about.

**2021.43 (08/08/2021)**

\* Add the new app's options Window - Position, which allows to choose if we want that our app's executables window appear centered in the screen or at some specific top and left positions. Before this change the app's executables window appears always centered, but now we can choose between this option and also a specific top and left positions.

\* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

**2021.42 (08/07/2021)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.1.0 version for the program's welcome page, program's help and program's about.

**2021.41 (07/17/2021)**

\* Add the new Download sample HTML app. This new sample app show how we can download arbitrary files from a server and then write into the file system. In the sample we download a sample.mp4 video file, however, it's possible to download any arbitrary file, just by changing the specified MIME type for the file. Nothing special is used, just the XMLHttpRequest Javascript class to download the file and the dhc.base64.decodeToFile() function provided by DecSoft HTML Compiler to write the file into the file system. The download can be started and aborted (if you want to allow it) by the user.



#### **2021.40 (06/23/2021)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.0.2 version for the program's welcome page, program's help and program's about.

#### **2021.39 (06/14/2021)**

\* Prepare the app's executables optional main menu (introduced in the previous release of the product) to be DPI awareness, that is, to scale the used items' images according to the Windows DPI configuration. This is useful to get the items' images properly scaled in 4K monitors, for example, or if the system screen in which our apps run is configured to work in sizes larger than the default 100% scale size.

#### **2021.38 (06/06/2021)**

\* Add the NavBar sample app, who uses the new app's main menu capabilities to set a kind of navigation bar with Back and Forward items, which the user can press in order to navigate back and forward in the browser's history.

\* Add the MainMenu sample app, who shows the usage of the `dhc.mainMenu.setItems()` external Javascript function, the `dhc.mainMenu.updateItem()` external Javascript function as well the `dhc.events.onMainMenuItemClick()` external Javascript function event.

\* Add the new `dhc.events.onMainMenuItemClick()` external Javascript function event. This function (if exists) is called when the user click in one of the main menu items that we can create with the `dhc.mainMenu.setItems()` external Javascript function.

\* Add the new `dhc.mainMenu.updateItem()` external Javascript function. This function is intended to update a main menu item after we create it. We can set a new caption, enabled state and image of a previously created main menu item.

\* Add the new `dhc.mainMenu.setItems()` external Javascript function. This function is intended to prepare a window's main menu for our app. We can establish any number of menu items and subitems, which can have his own "caption", "id" and "image". See the help for more information and details.

\* Enhance the app's Javascript option window in order to resize the Javascript editor to fit the window size, so it's more easy for us to prepare and write the Javascript code that we want to place in that app's Javascript option.

### **2021.37 (05/22/2021)**

\* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

### **2021.36 (05/20/2021)**

\* Add into the program's help the right documentation and sample code for the recently added "dhc.system.\*" external Javascript functions.

\* Add the new System sample app, who show how to use the recently added "dhc.system.\*" external Javascript functions.

\* Add the new dhc.system.shutdown() external Javascript function. This function is intended to shutdown the Windows system from our apps.

\* Add the new dhc.system.reboot() external Javascript function. This function is intended to reboot the Windows system from our apps.

\* Add the new dhc.system.logoff() external Javascript function. This function is intended to logoff the Windows user session from our apps.

### **2021.35 (05/10/2021)**

\* Update the Bootstrap CSS framework (including the CSS and Javascript) to the latest 5.0.0 version for the program's welcome page, program's help and program's about.

### **2021.34 (04/16/2021)**

\* Add the new dhc.folders.getAppDataDir() external Javascript function. This function is intended to get the directory path intended to save app's related files. We can also use the dhc.folders.getUserDesktopDir(), for example, to store files, but, this last directory is mainly intended to save app's user's related files.

\* Remove an annoying flicker in the main toolbar Save button (basically the button alternates between the enabled / disabled states without apparently any reason). The button work as expected without any problem, however, the flicker referred to can certainly be a bit annoying distracting us about what we are doing.

\* Add a new button in the program's behaviour options in order to delete the recent files list. Make also some other internal changes in order to properly show the right images for the window of the program's dock panels (when floating), like the program's options,

welcome page, about, and other dock panels.

### **2021.33 (04/15/2021)**

- \* Add the new Signature sample app, which show to us how to use the recently added `dhc.base64.decodeToFile()` external Javascript function in order to save a canvas control blob into a PNG image file.
- \* Add the new `dhc.base64.decodeToFile()` external Javascript function. This function is intended to save base64 strings into files after decode it. This can be useful, for example, to save Javascript blob objects, like a canvas control blob object into images. Thanks **Leonardo** for the idea!

### **2021.32 (04/10/2021)**

- \* Prepare the Microsoft Edge Runtime to enable the Web Speech API. This API is quite experimental but interesting. You can find some examples of usage and get some fun from this Mozilla page: [https://developer.mozilla.org/docs/Web/API/Web\\_Speech\\_API](https://developer.mozilla.org/docs/Web/API/Web_Speech_API) Thanks **Leonardo** for the idea!

### **2021.31 (04/07/2021)**

- \* Appropriately handle app's files which contain spaces in their names: before this change / release the app's files with spaces cannot properly work if they are included in the app's executables. Thanks **Greg** for the report!
- \* Fix a minor bug which causes that the icon which appear in the New app's index file dialog becomes the "help" instead of the "folder" when we over the mouse by the index file input or just select a new index file for the new app.

### **2021.30 (04/05/2021)**

- \* This is the first major release of DecSoft HTML Compiler, means if you purchase the product before 04-04-2020, you must to purchase an upgrade license (with a 50% off discount), or download the outdated release to continue using your outdated license. If you purchase your license after the referred date, please, login into your DecSoft customer area to grab your new license serial number. You can purchase an upgrade license as well to download the outdated release from your DecSoft customer area in our website.

#### **2021.29 (04/04/2021)**

- \* Enhance the program's help to describe and provide samples of code for the new `dhc.themes.*` related external Javascript functions, which allows to deal with the app's executable themes at runtime.
- \* Add the new Themes sample app, which shows how we can use the new `dhc.themes.*` related external Javascript functions, in order to get the list of the available themes for the app's executable, get the current established theme and set a specific theme at runtime.
- \* Add the new `dhc.themes.getThemes()` external Javascript function, which we can use to retrieve a list of the available themes to be set for the app's executable.
- \* Add the new `dhc.themes.getTheme()` external Javascript function, which we can use to know the current app's executable theme name. We can set a theme to be used at designtime, from the app's options. We can also set the app's executable theme at runtime by using the `dhc.themes.setTheme()` external Javascript function.
- \* Add the new `dhc.themes.setTheme()` external Javascript function, which allows to establish the app's executable theme at runtime, in addition to use the app's options, which allows to establish the app's executable theme at designtime.

#### **2021.28 (04/01/2021)**

- \* Add the new FindWindow sample app, which show how we can use the `dhc.shell.windows()` in order to determine if a specific program is running in the user's computer, based in the program's windows' captions.
- \* Add the new `dhc.shell.windows()` external Javascript function. This function retrieves all the windows' captions from the executing programs in the user's computer. This function can be useful, for example, do determine if a specific program is running or not, based in the program's windows' captions.

#### **2021.27 (03/31/2021)**

- \* Add the new `dhc.folders.getAppExecutableDir()` external Javascript function, which let us to know the absolute directory path of the app executable. This function can be useful to get that directory path, and to conform absolute paths for certain files, in order to be used with other Javascript functions like `dhc.shell.command()`, for example.

#### **2021.26 (03/28/2021)**

\* In this product release we make several internal changes and enhancements in order to prepare the GUI (Graphic User Interface) to be ready for 4K monitors, and, in general, for screens with a DPI (Dot Per Inches) configuration larger than the used when designing the product, that is, the Windows default DPI configuration. The menu and toolbars icons, the dialogs and all the GUI forms and controls have been reviewed and tested to achieve this objective. We are particularly proud of this product release, because the work we expend on it and because the results that we finally can obtain.

\* We experience certain very, very rare bugs due to the lack of initialization for certain variables related with some events of the Edge Runtime WebView. This product release fixes these rare bugs by properly initializing the appropriate variables before trying to use it.

#### **2021.25 (02/20/2021)**

\* The program Welcome Page has been redesigned in order to be more responsive for different sizes and now additionally show more latest forum threads and latest forum posts as well the latest entries from the DecSoft blog. No other changes has been made for this release of the product.

#### **2021.24 (02/12/2021)**

\* We are particularly happy with this new release of the product, since we can finally fix an annoying error (difficult to reproduce, difficult to fix) which sometimes occur when compile the app's executables. We not only fix the error, but, also try to catch a possible similar error in another way, informing to the user in a more properly (in the compilation log) and avoiding the default system exception message dialog.

\* Avoid (fix) some problem which can occur if we fastly press the compilation button twice: commonly this is something that we don't do, however, until this release can be possible to do it and therefore an error occur. This release fix this problem by disabling the compilation button once we press it.

#### **2021.23 (01/16/2021)**

\* Add also the new Video sample app, which show how we can include videos in our apps embedding it in our app executables or not. Take a look at the Video sample app, since you can see that the "autoplay" or the "preload" attributes for the video HTML tag are required in order to properly works.

\* Add the new Audio sample app, which show how we can include audios in our apps embedding it in our app executables or not. Take a look at the Audio sample app, since you

can see that the "autoplay" or the "preload" attributes for the audio HTML tag are required in order to properly works.

- \* Update the Microsoft Edge Runtime installer to the latest version. This update includes all the supported flavors, that is, the full installer for both 32 and 64 bits and the downloadable installer for both 32 and 64 bits.

### **2021.22 (01/02/2021)**

- \* Update the DecSoft HTML Compiler Welcome page and HTML help in order to use the latest version of the Bootstrap CSS framework. This change do not affect to your apps, only internally to the IDE in the referred Welcome page and HTML help.

### **2021.21 (11/21/2020)**

- \* Add the new `dhc.shell.command()` Javascript external function, which can be used to execute a Windows command line and retrieves the produced output. Take a look at the updated product's help and also to the new Comand sample app.

### **2021.20 (11/20/2020)**

- \* Add the new Javascript external function `dhc.window.setTitle()`, which allows to set the title of the app's executable window at runtime. Look at the Window sample app, as well the updated product's help, for more information.

- \* Add the new Javascript external function `dhc.window.getTitle()`, which retrieves the current title of the app's executable window. Look at the Window sample app, as well the updated product's help, for more information.

- \* Add the new app's Compilation option "Change the app's window title using the HTML title tag". This option is unchecked by default (we use then the app's name for the window title), but can be checked if you want to change the app's executable window title according to the HTML title tag of the current HTML file.

### **2021.19 (11/15/2020)**

- \* Add the new Javascript external function `dhc.shell.explore()`, which allows to explore (show it in the Windows explorer) a user's system file or folder. Look at the new Shell sample app, as well the updated product's help, for more information.

- \* Add the new Javascript external function `dhc.shell.execute()`, which allows to execute a

file with the system's default program for that kind of files. Look at the new Shell sample app, as well the updated product's help, for more information.

#### **2021.18 (11/14/2020)**

- \* Add the new Javascript external function `dhc.files.extractFile()`, which allows to extract a file (compiled in our app's executable) into the user's file system. Look at the new ExtractFile sample app, as well the updated product's help, for more information.

- \* Add the new Javascript external function `dhc.window.setState()`, which allows to set the state (normal, minimized or maximized) of the app's executable window. Look at the new Window sample app, as well the updated product's help, for more information.

- \* Add the new Javascript external function `dhc.window.getState()`, which retrieves the state (normal, minimized or maximized) of the app's executable window. Look at the new Window sample app, as well the updated product's help, for more information.

- \* Add the new Javascript external function `dhc.window.setStyle()`, which allows to set the style (normal or on top) of the app's executable window. Look at the new Window sample app, as well the updated product's help, for more information.

- \* Add the new Javascript external function `dhc.window.getStyle()`, which retrieves the style (normal or on top) of the app's executable window. Look at the new Window sample app, as well the updated product's help, for more information.

#### **2021.17 (11/06/2020)**

- \* Add the new State option in the app's options -> Window. This option allows to start the app in a normal state (by default) or in a maximized state, similar to the full screen style, but, maintaining the app's window title bar available.

#### **2021.16 (10/27/2020)**

- \* The new windows of your app's files are internally handled by DecSoft HTML Compiler in order to maintain one of the main characteristic of the program: never extract your app's files in the user computer. However, in rare situations (basically when we are dealing with an external website, not our app's fiels) the handle can cause some troubles with new windows: this release continue handling the new windows of your app's files as before, but, let the browser's webview2 to handle new windows which point to external websites URLs, to avoid the referred problems when handle this kind of URLs too.

#### **2021.15 (10/26/2020)**

- \* Modify the Website sample app in order to use the new app's Javascript option. What we do is to toggle the app's window full screen even in the case that we redirect the user to an external website, out of the scope of our app's files. So the Website sample app can toggle the full screen now by pressing the F11 key, just by executing the code that we place in the app's Javascript option.
- \* Add to the app options the new Javascript tab. This new option allows to place some Javascript code to be executed when the browser's webview is loaded. This option is to be used in rare situations, like if we redirect the user to an external website: then the Javascript code that we can place in our app's "index.html" are not executed, but the code of this new option is executed even in this case.

#### **2021.14 (10/23/2020)**

- \* Add to the app options a new Signing tool tab, which allows to select one of the configured signing tools from the program options. You can select in this new options tab the signing tool to be used to digitally sign your app's executables when compile it.
- \* Add to the program options a new Signing tools tab, which allows to establish one or more code signing tools in order to be used to digitally sign your app's executables when compile it. You can add one or more signing tools command line strings in order to be used later when compile your app's executables.
- \* The app options window tabs has been reordered for a better accesibility, and these changes has been properly updated in the program help file.

#### **2021.13 (10/22/2020)**

- \* Add the ability to use the Microsoft Edge Runtime download installer. We can know choose to include in our app's executables the Microsoft Edge Runtime full installer (no internet connection required but a greater size) or the Microsoft Edge Runtime download installer (internet required but a smaller size).

#### **2021.12 (10/20/2020)**

- \* Update the Microsoft Edge Runtime installer to the latest version. The latest version of the installer has been published in different flavours for Windows 32 and 64 bits, so, DecSoft HTML Compiler also include this two flavours and use (if we want to include the installer)



the right one for the produced executables for Windows 32 and 64 bits.

### **2021.11 (10/17/2020)**

\* Add the new sample app Pacman, which wants to show how easy we can compile into Microsoft Windows executables for 64 and 32 bits an HTML5 game, including audios, etc. The game is an open source development made by <http://www.putchu.be>. Other minor changes, fixes and enhancements has been made for this new release of the product.

### **2021.10 (10/05/2020)**

\* Only if the user's computer doesn't have the Microsoft Edge WebView2 Runtime installed, our compiled app's executables can do two things: show to the user a message dialog indicating that the app cannot continue, or, if our app include the Edge Runtime installer, invite the user to install it with another message dialog. For these referred dialogs we are using the english language until now, but now DecSoft HTML Compiler check the user's system language and show these messages in the user's language, if supported, or fallback to the english language. The supported languages for now are spanish, french, italian, german and chinese.

### **2021.9 (10/03/2020)**

- \* Update the Microsoft Edge WebView2 Runtime installer to the latest version.
- \* Replace the IFrame sample app with the Website sample app. The point of these samples is to use DecSoft HTML Compiler to "compile" a website into a standalone executable for Microsoft Windows. Some websites can refuse to be load inside an Iframe HTML element, so, the new Website sample app take another approach: redirect to our desired website using a bit of Javascript in our app's index.html file.

### **2021.8 (09/30/2020)**

- \* Add a new app compilation flags option to allow or disallow (by default) multiple instances of the app's executables. By default the app's executables can't run twice, that is, don't allow multiple instances of the app running at the same time. Thanks **Kevin** for the idea!
- \* Fix a bug which causes that we can't use the "Explore app folder" action in the program's main window toolbar in certain circumstances, for example, after save the app's options. Now this action works as expected, exploring the app's folder in any circumstances.

#### **2021.7 (09/28/2020)**

- \* Now the app's executable window title bar uses the HTML document title instead of the app's name (which is still used by default): this can help in order to differentiate opened windows of the same app. Thanks **Kevin** for the idea!
- \* Fix a bug which causes that the debugger developer console still available to be shown even if we choose not to shown it from the app's options. Now the option that we set are respected as expected. Thanks **Kevin** for the report!

#### **2021.6 (09/27/2020)**

- \* Add the new `dhc.app.inFullscreen()`, `dhc.app.enterFullscreen()` and `dhc.app.exitFullscreen()` external Javascript functions, which allows to know if our app is running in fullscreen mode, enter in fullscreen mode and exit from the fullscreen mode. Take a look also at the new Fullscree sample app.

#### **2021.5 (09/26/2020)**

- \* We move our products from `davidesperalta.com` to `decsoftutils.com` and therefore some URLs has been changed into this release of DecSoft HTML Compiler. No more other changes has been made for this release of the product.

#### **2021.4 (09/25/2020)**

- \* Added a new Information tab into the app's options which allows to establish optional information for the app compiled executables: product name (already used app name), company name, file description, file version, internal name, legal copyright, legal trademarks, original file name, program ID, product version and comments.
- \* Allow to select another file different than an "index.html" one for the app's index file. This works now for both design time and runtime without problems, so our app index file can be "anyother.html" and not mandatory "index.html".

#### **2021.3 (09/23/2020)**

- \* Add the new Close sample app, which shows how we can react when the user wants to close the app, but also uses all the new Javascript external stuff which has been added in

this release of the product.

- \* Add the new `dhc.app.close()` external Javascript function, which allows to close the app programmatically. Take a look at the recently added Close sample app.

- \* Add the new `dhc.events.onCloseQuery()` external Javascript function / event, which you can use to react when the app goes to be closed by the user, for example. Take a look at the recently added Close sample app.

- \* Add the new `dhc.isRuntime()` external Javascript function in order to know if our app is running at design time (opened in DecSoft App Builder) or if it's running alone in a compiled executable. This can be useful sometimes, for example, you can take a look at the recently added Close sample app.

- \* The permissions to access the Microphone, Camera, Geolocation, Notifications, Generic Sensors and the Clipboard Read are automatically granted now, avoiding the annoying default browser's dialog asking to the user.

## **2021.2 (09/22/2020)**

- \* Add the new `dhc.folders.getTempDir()`, `dhc.folders.getUserDocumentsDir()` and `dhc.folders.getUserDesktopDir()` external Javascript functions. For more information you can take a look at the app's external Javascript documentation and also at the Files sample app.

- \* Reorganize the `dhc.*` external Javascript functions into categories, right now files, folders and dialogs. We do that in order to avoid possible identifiers collisions and also for a better understanding of the available external Javascript functions. The documentation and the Files sample app has been updated accordingly.

- \* Fix a bug in the external Javascript `dhc.deleteDir()` function, which indicates to use a "dirPath" argument, but in fact expected a "filePath" argument. Now the function expected the same than the documentation establishes: a "dirPath" argument.

## **2021.1 (09/20/2020)**

- \* After months of hard work we are proud to publish the first public stable release of the new generation of DecSoft HTML Compiler. We hope this journey just begins and we can enhance this product with a lot of future releases.

For current DecSoft HTML Compiler customers, the below post text can be interesting, since it compares a bit the previous generation and the new generation of DecSoft HTML Compiler:

1º Say good bye to the venerable Microsoft Internet Explorer WebView and say welcome to the modern Microsoft Edge WebView2 based in Edge Chromium. The DecSoft HTML

Compiler generated apps continue to be running standalone, with the only dependency of the Microsoft Edge WebView2 Runtime, and, the ability to install it if it's missing in the user's computer and you wanted to do it from the app itself.

2º Say also good bye to previous limitations, for example, in order to make AJAX request to your app's local files, play local audios and local videos, etc. With the new DecSoft HTML Compiler all of these is possible, and, for example, applications created by DecSoft App Builder can perfectly be compiled using the new DecSoft HTML Compiler and run like a charm.

3º Like the current generation of DecSoft App Builder, the new DecSoft HTML Compiler will be offered for both Microsoft Windows 32 and 64 bits. Additionally, the new DecSoft HTML Compiler can compile apps into executables for Microsoft Windows 32 and 64 bits (not only for 32 bits like now): it's even possible to create executables for Microsoft Windows 64 bits from Microsoft Windows 32 bits.

4º The new DecSoft HTML Compiler has been written from the scratch, applying the experience that we get while developing the current DecSoft App Builder. This have lot of beneficts, just to mention some of them, now it's possible to work with various HTML apps at the same time in the program, we can now cancel the compilation of an app, we avoid modal dialogs always that it's possible and many, many more characteristics.

So I am very excited with this new project, because the ability to compile DecSoft App Builder created apps using DecSoft HTML Compiler, because the use of the new Microsoft Edge WebView2 based in Edge Chromium. However, the new DecSoft HTML Compiler wanted to continue offering the features that make to the current DecSoft HTML Compiler a good product:

1º Your app's files are never extrated into the user computer. The generated app's executables (now for both Windows 32 and 64 bits) is the only file that your users needed to run your app, no other dependency is required. You can continue using dozens of different GUI themes in your app's executables. In few words, we wanted to maintain the best that the current DecSoft HTML Compiler can offer right now.

## **Download and contact**

[www.decsoftutils.com](http://www.decsoftutils.com)

<mailto:info@decsoftutils.com>