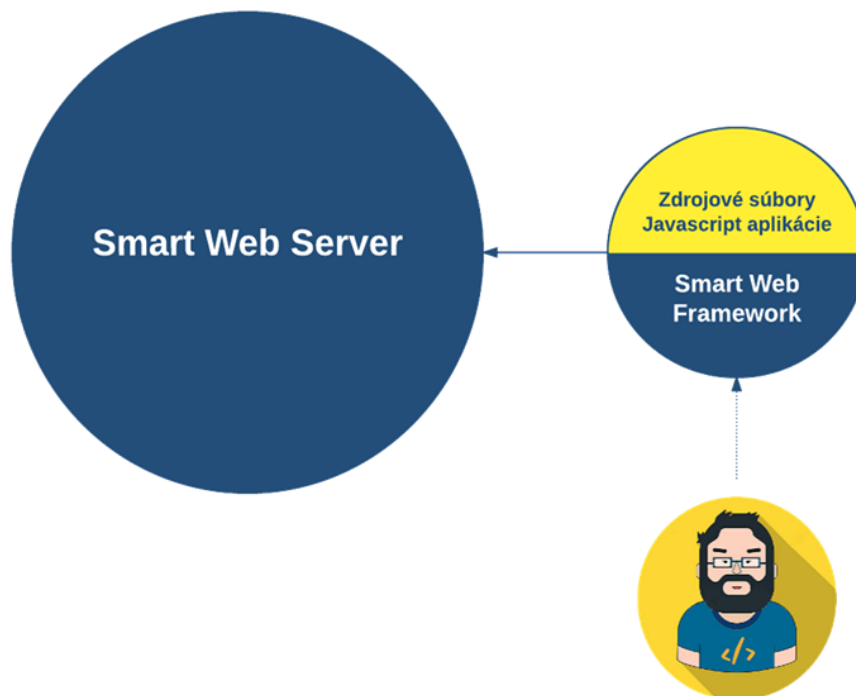


# SmartWeb Framework Library

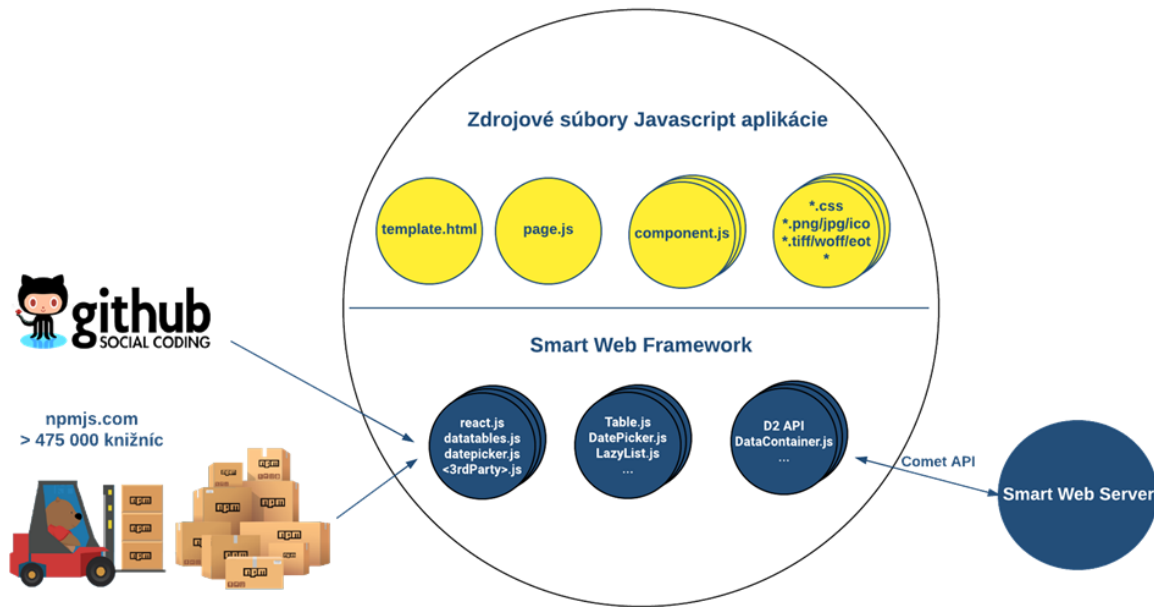
Another important function of the SmartWeb platform is a special Javascript library designated for the web applications' development on D2000. An application programmer can use this library or its parts while implementing individual application pages.



The SmartWeb Framework (SWF) library serves mainly for making easier the work of an application programmer, who than does not have to *reinvent the wheel* in the development process. The way of using the SmartWeb Framework library reflects the recommended way of developing web applications in the Ipesoft company. This library is completely optional for the development of web applications. Thus a programmer can implement an application without using it. SWF implements *basic functions* which every web application uses, first of all:

- communication with D2000
- displaying of UI components

Communication with D2000 is implemented in a code that realizes a connection to the universal Comet interface of the SmartWeb server. Besides communication with the server, several often-used non-trivial UI components (e.g. table, list box and so on) are implemented in the library. These components are not implemented by the D2000 Team responsible for SmartWeb as a project, but they are components from third-party libraries imported to the application using the [Bundle Engine](#) module. The following picture shows individual parts of the web application which uses the SmartWeb Framework.



In the upper part of the picture, there are displayed files created by an application programmer:

- The file `template.html` is a template of an HTML file which is sent to a browser for every web application page. When requesting particular page of a browser, the SmartWeb Server completes this template by adding a link to a file with a code of the page itself. For example, when requesting the page `page.html`, also the file `page.js` shows up which contains a code of this page.
- Application codes of several pages can use some common components specific for this application. The aim of the shared component (displayed in the picture as `component.js`) is to prevent code redundancy.
- Besides an HTML template and a page code, it is possible to add web artifacts such as CSS styles, pictures, fonts, videos and so on.

In the lower part of the picture, there are displayed following parts of the SmartWeb Framework which the application can use:

- Third party libraries such as [React JS](#), [Datatables](#) and so on. The list of used libraries can be extended by any other library published in a public repository of Javascript libraries [npmjs.com](#) or from the repository of the open-source code [github.com](#).
- Further, SWF contains often-used UI components (`Table.js`, `DatePicker.js`) programmed for the React framework (the reason for choosing this library will be explained later). Behind every UI component, there is a particular open-source library used which in reality implements the given component. This means that every UI component is only so-called *wrapper* above the third party library. This wrapper publishes the library's API, or else extends or repairs its behaviour and enables trouble-free usage of components in the [React JS](#) framework.
- The last group of functionalities provided by the SmartWeb Framework is related to data acquisition from D2000 via the universal API SmartWeb server. The class `D2Api` implements communication with D2000 via the universal Comet interface of the SmartWeb server. This class is independent of the used UI framework and it is possible to use it individually. Except for `D2Api`, an application programmer can make `D2 API` and React framework accessible by a [concept of a data container](#) (in our case, it is the `DataContainer` class).

The UI components of the SmartWeb Framework are built on the [React JS](#) library from the Facebook company. The reasons for the use of the React JS framework for creating web applications are the following:

- Currently, it is the most often used framework for creating the UI interface; [from a survey carried out in 2017](#) on a sample of around 24 000 web programmers, the following findings emerged:
  - 58% of respondents use the React JS framework for creating web applications and at the same time they want to use it in a next project
  - 27% of respondents have heard of it and they want to learn it.
- The *mature* technology is integrated into [many of the most often-used applications](#) (facebook.com, pinterest.com, dropbox.com, uber.com, ...).
- Since it is widely used, there are many available components, detailed documentation, and descriptions of problem-solving.
- There is a low threshold for learning considering the framework possibilities - it is possible to mentally comprehend the concept and API by [reading one article](#).
- There is a possibility to declare the design of the application site similarly to HTML in 99% - [in the JSX format](#) usable directly in the Javascript code.