SAGE2 User documentation

- Introduction to SAGE2
- Starting SAGE2
  - server
  - Display Clients
  - Audio Manager
  - Web Interface
- Loading content
  - Supported File Types
  - Desktop Sharing
  - Applications
  - Sessions
- Interacting with content
  - Web interface
  - Mouse and keyboard operations & shortcuts
- Configuration
    - Configuration Page
  - SAGE2 config
    - Minimal Sample
    - Expanded Sample
- Applications
  - PDF Viewer
  - Movie Player
  - Image viewer
  - Desktop sharing
  - stereo_image
  - UIC_crime_with_leaflet
  - photos
  - US_weather
  - radar

car_threejs
chronicles_of_spaceman_spiff
sticky_note
clock_svg
web_earth
flow
welcome
googlemaps
zoom
notepad

1. Introduction to SAGE2

1.1. Introduction
1.1.1. this guide
1.1.1.1. target audience / requirements
1.1.2. Getting help
1.1.3. Getting the software
1.2. Basics of SAGE2
1.3. SAGE2 structure
1.4. Graphical interface

2. Starting SAGE2

2.1. server

The SAGE2 server is an enhanced web-server written as a Node.js application, started by a command similar to:

- node server.js [options]

In the binary distribution, the execution of SAGE2 is controlled by a 'launcher' process (called *sabi.js*). To start the launcher, double-click *'Launcher.bat'.* The launcher is itself a web server, which can be accessed remotely for remote management. The launcher is running at the following URL: http://localhost:1000. A chrome window is opened automatically to the launcher. The basic functions are accessible at http://localhost:1000/#SAGE2 The launcher is protected by a password. Default values are, name: *sage2* and password: *sage2.* The password can be changed in the *Admin* page. The basic functions are accessible at http://localhost:1000/#SAGE2 : Start a SAGE2 session, stop a SAGE2 sessions, and set a meeting access code (password for end-users to access the SAGE2 session).
The admin page (http://localhost:1000/#Admin) is accessible by pressing the button at the bottom of the page. One can edit the SAGE2 wall configuration using either a web form (basic edits) or a JSON file editor (expert edits). To change the launcher password, enter your new password in the text box (in the form *sage2:sabi:newpassword*) and click 'Reset launcher password'. The password will be activated at the next start of the launcher

The server will list the URLs for each client type in the console.

2.2. Display Clients

Once the server is running, display clients are accessible at the URL:

- http:// [server IP] : [port] /display.html?clientID=[client number]
- ex: **http://localhost:9090/display.html?clientID=0**

2.3. Audio Manager

Once the server is running, the audio manager is accessible at the URL:

- http:// [server IP] : [port] /audioManager.html
- ex: **http://localhost:9090/audioManager.html**

2.4. Web Interface

Web interface used by end-users need to use the secure connection URL to access SAGE2. If connecting to the unsecure port, users will be redirected automatically to the secure connection.

- https:// [server IP] : [port] /
- ex: https://localhost:9090/
- ex: https://myserver.edu/

3. Loading content

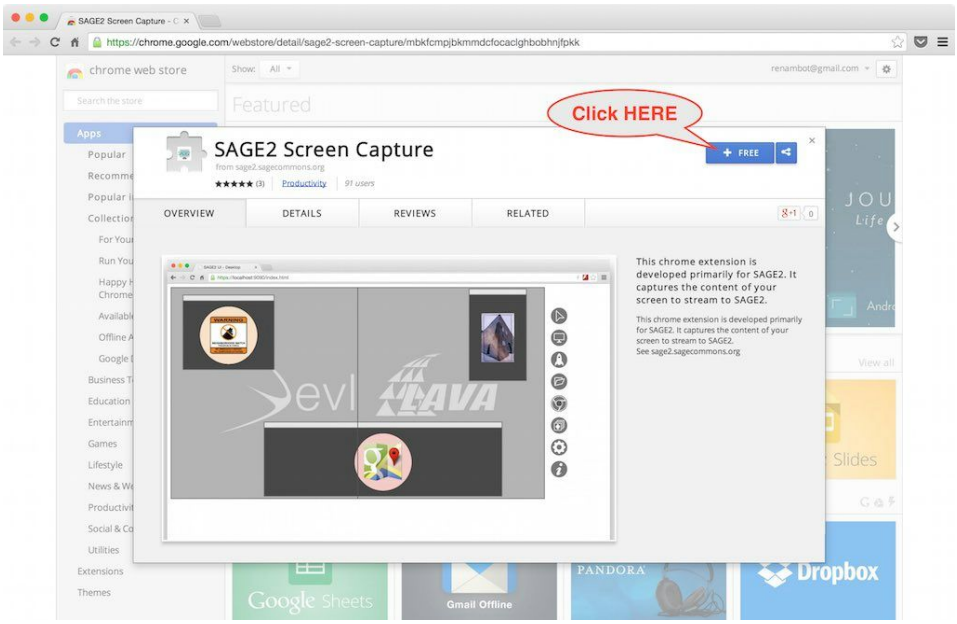
3.1. Supported File Types

- Images
  - JPEG, PNG, TIFF, BMP, PSD
- Videos
  - MP4, M4V, WEBM, MOV (containing MP4)
- PDFs
  - files with many layers might be an issue sometimes (output from Illustrator, InDesign, …). When possible, “optimize” the document for a web output.
- 3D images:
  - side by side images (full width): PNS and JPS

3.2. Desktop Sharing

Desktop sharing is supported using using a Chrome Extension inside the browser. You will be requested to install it at the first use.
The first time before using SAGE2, one must enable screen capture support in Chrome by installing a Chrome extension: Navigate to the Chrome Webstore for extensions and install the SAGE2 Desktop Sharing extension (click on the + sign): SAGE2 Screen Capture:

- *https://chrome.google.com/webstore/detail/sage2-screen-capture/mbkfcmpjbkmmdcfocaclghbobhnjfpkk*



3.3. Applications

We support a Javascript API to build applications. A few applications are included in the distribution. More applications are available as example is a separate repository.

- **Sample applications**
    - Beside the applications provided when installing, a repository of sample applications is available at:
        - https://bitbucket.org/sage2/sage2_apps
        - see javascript sub-folder
    - Samples
        - atom_smasher: webgl demo
        - bounce: canvas 2d app
        - cave2_monitor: D3.js dashboard for CAVE2
        - clock: canvas 2d app
        - d3_sample: a D3.js example
        - evl_photos: slideshow
        - evl_weather: D3.js weather map in the lab
        - forecast: weather app using forecast.io
        - kinectic_animation: kinectic.js demo
        - kinectic_oscillating: kinectic.js demo
        - modest: Modest map API demo
        - playcanvas: demo of game engine Playcanvas
        - presentation: iframe to office365
        - prime: calculate prime number in a web-worker
        - radar: radar map in D3: click to cycle through locations
        - snap_one: demo in snap.svg
        - stereo_viewer: another stereo image viewet, in webgl

- tank: multiple player game in canvas
- texture_cube: webgl demo
- three_sample: three.js demo
- three_shader: three.js demo
- tweetcloud: from twitter to word cloud
- twitter_frame: twitter stream into iframe
- unity: loading unity demo with plugin into an iframe
- widget_demo: demo of SAGE2 widget API

3.4. Sessions

4. Interacting with content

4.1. Web interface

SAGE2 Pointer has two modes: window management (arrow plus cross pointer) and application interaction (arrow pointer). You activate the pointer by pressing the arrow icon on the web interface. The first time, you need to allow Chrome to capture mouse events from the browser.

- Window Management Mode
  - Left-click anywhere on an application to move window
  - Left-click bottom-right corner of an application to resize window
  - Right-click on an application to bring up widget bar
  - Right-click on an widget bar to hide it
  - Scroll anywhere on an application to resize window
- Application Interaction Mode
  - All events forwarded to the application
  - Press 'Shift+Tab' to toggle SAGE2 Pointer modes
  - Press 'Esc' to release the SAGE2 Pointer
  - Move the window by clicking and dragging on the title bar
  - Resize the window by clicking and dragging the bottom right corner

4.2. Mouse and keyboard operations & shortcuts

SAGE2 Cheat Sheet – Mouse and keyboard operations & shortcuts

Pointer modes	
Application	Application interaction mode: directly control applications with keyboard and mouse (*titlebar* and *corner* active)
Window	Window interaction mode: move and resize windows
Shit-TAB	Switch modes
ALT	Temporary switch (while pressed)
Spacebar	Activate the pointer
ESC	Exit pointer mode

Window Interaction	
Left button	Move/Resize window
Right button	Show/Hide widgets
Wheel (scroll)	Resize window (2 finger scroll gesture)

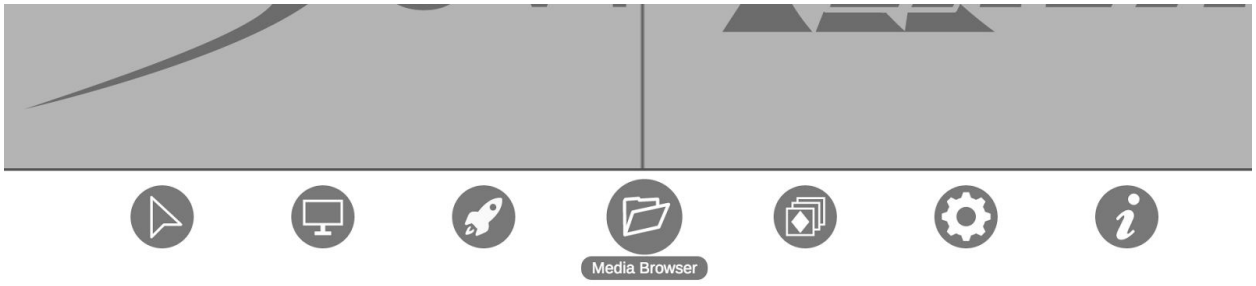
Video Player	
Spacebar	Play/pause
l	Toggle looping
m	Toggle mute
r	Rewind

PDF Viewer	
Spacebar	Next page
Left arrow	Previous page
Right arrow	Next page
1	First page
0	Last page
r	Refresh

Image Viewer	
i	Show EXIF data

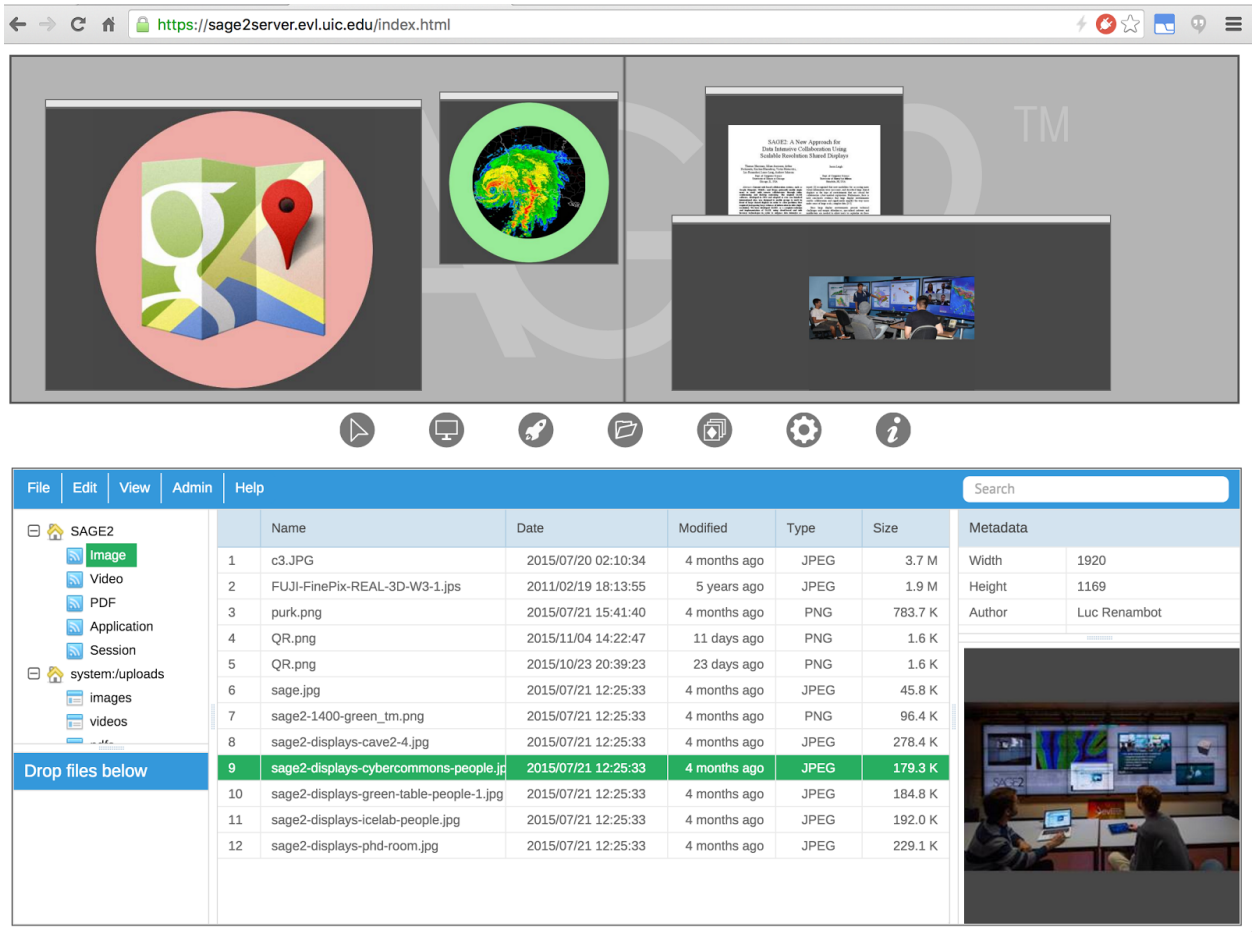
Google Maps	
Left button	Drag to pan
Mouse wheel	Zoom in and out
Left/Right/Up/Down	Pan the map
m	Loop through map types
t	Toggle traffic layer
Alt/Ctrl	Zoom in/out

Zoom (scalable) image viewer	
Left button	Drag to pan
Mouse wheel	Zoom in and out
Left/Right/Up/Down	Pan the map
Double left click	Zoom in
Shift + Double left click	Zoom out



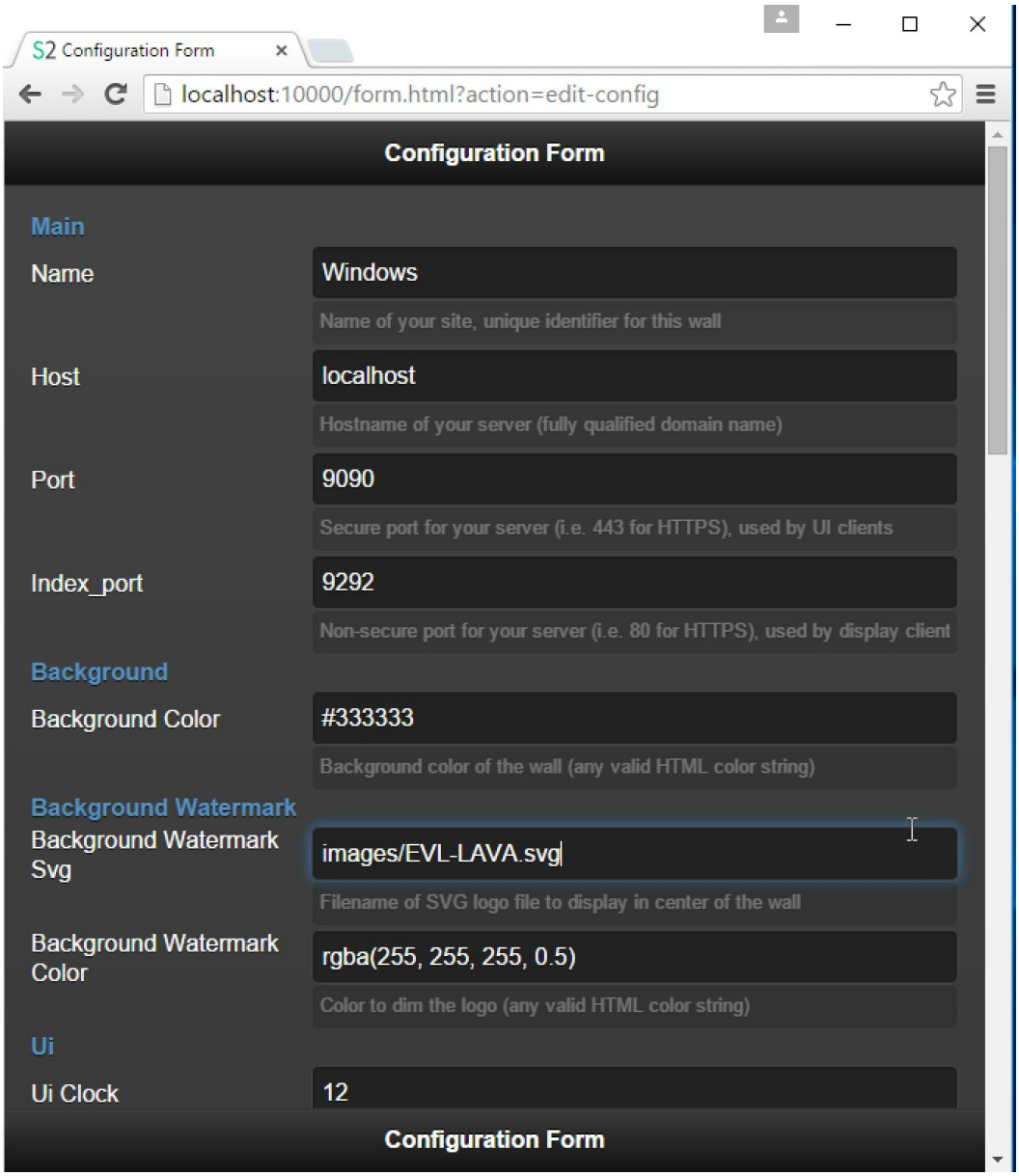
A file manager is provided inside the web interface. Click the

- start application
- open content (pdf, video, images, …)
- browse the several media folders
- organize content in folders
- delete content
- upload new files
- download a local copy of the any asset


https://sage2server.evl.uic.edu/index.html
File
Edit
View
Admin
Help
Search
SAGE2
Image
Video
PDF
Application
Session
system:/uploads
images
videos
Drop files below
Name
Date
Modified
Type
Size
1 c3.JPG 2015/07/20 02:10:34 4 months ago JPEG 3.7 M
2 FUJI-FinePix-REAL-3D-W3-1.jps 2011/02/19 18:13:55 5 years ago JPEG 1.9 M
3 purk.png 2015/07/21 15:41:40 4 months ago PNG 783.7 K
4 QR.png 2015/11/04 14:22:47 11 days ago PNG 1.6 K
5 QR.png 2015/10/23 20:39:23 23 days ago PNG 1.6 K
6 sage.jpg 2015/07/21 12:25:33 4 months ago JPEG 45.8 K
7 sage2-1400-green_tm.png 2015/07/21 12:25:33 4 months ago PNG 96.4 K
8 sage2-displays-cave2-4.jpg 2015/07/21 12:25:33 4 months ago JPEG 278.4 K
9 sage2-displays-cybercommons-people.jp 2015/07/21 12:25:33 4 months ago JPEG 179.3 K
10 sage2-displays-green-table-people-1.jpg 2015/07/21 12:25:33 4 months ago JPEG 184.8 K
11 sage2-displays-icelab-people.jpg 2015/07/21 12:25:33 4 months ago JPEG 192.0 K
12 sage2-displays-phd-room.jpg 2015/07/21 12:25:33 4 months ago JPEG 229.1 K
Metadata
Width 1920
Height 1169
Author Luc Renambot

5. Configuration

Through the launcher, basic configuration of the wall is possible. A advanced editor is also available for expert users.



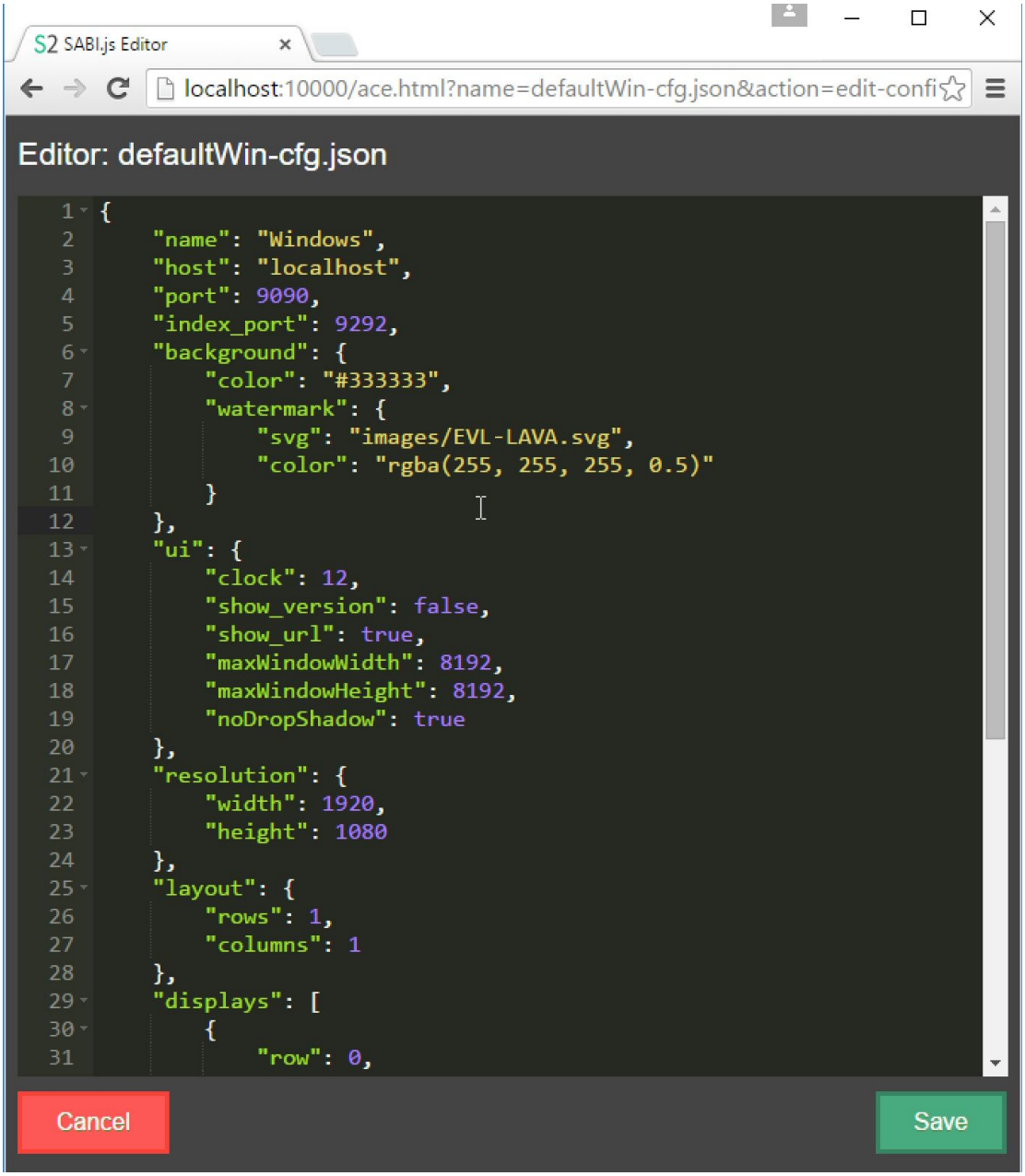
Configuration Page

- **Hostname / IP to display**: SAGE2 shows this value in the top left corner of the display. This will be visible for all using the SAGE2 wall. If you do not have a static IP, you may need to frequently update this value.
- **Alternate Hostnames / IPs**: Add all hostnames and IP's from which users should be able to access the SAGE2 wall in addition to the displayed value. Even if the machine is

associated with other IP addresses, only the values listed will be allowed for users to connect in on.

- **Port number**: By default this is 9292. But will require everyone who connects to your wall to add :9292 to the address in order to access. To use standard http port, use 80. This will remove the need to specify the port number. Secure Port Number: By default 9090. By using 443, users connecting to the SAGE2 wall will not need to specify the secure port.
- **Resolution of display**: This value must be in pixels. Do not include suffix or units marker. Layout: Number of panels in the display configuration. * Note: If you intend to use the web controller, the entered resolution value should be the entire width and height of all panels and for the panel layout it should be 1x1. The reason is that the web controller will launch one firefox window large enough to span the entire display.
- **Remote Sites**: Enter in sites that you wish to see the status and share files with. Label: The name to display on the display. Hostname / IP address: Valid hostname / ip which the remote site can be accessed. Secure Port Number: The port number that the remote site uses.
- **Dependency location on the computer**: Where the installations for ImageMagick and FFMpeg are installed. If you are using the SAGE2 binary zip, then this should always be bin\
- **Background**: A background color can be entered as a CSS hex value. If an image is desired to be shown, the path from the public directory of SAGE2 can be used. The path must be within the public directory otherwise the file cannot be served. The Option to use an SVG is also available, this too must use a public directory path reference otherwise it cannot be served.
- **Meeting ID**: When users connect to the SAGE2 wall they will be prompted for the meeting ID before being able to see content. This applies to the UI, display, and audio manager. Web Controller Password: Necessary if you want to use the web controller to remotely start, stop and reset the meeting ID. Configuration Page Password: Necessary to access the configuration page again.

One can also use the advanced mode to edit freely the generated configuration file. This enabled to add or delete configuration items, such as remote sites, extra displays, and experimental APIs (oMicron interaction server, …).



SAGE2 config

A JSON file to configure the display environment

```
{
    host:                       // hostname or ip address of the web server
    name:                       // OPTIONAL: name used in some UI (free string, defaults
                                       to hostname above)
    port:                       // HTTPS port that all clients are served on (default: 443)
```

```
    index_port:                     // HTTP port, provides a reroute to HTTPS (default: 80)
    rproxy_port:                    // OPTIONAL: port of the HTTPS reverse proxy (only required
                                           for reverse proxy setups)
    rproxy_index_port:              // OPTIONAL: port of the HTTP reverse proxy (only required for
                                           reverse proxy setups)
    url:                            // OPTIONAL: URL shown on display, instead of using hostname
                                            (allows for short URL)
    background: {
        color:                      // CSS color for the background (hex, rgba(), etc.)
        image: {                    // OPTIONAL:
            url:                    // relative path from the public directory to an image used
                                             for the background
            style:                  // either "fit", "stretch", or "tile"
        }
        watermark: {                // OPTIONAL:
            svg:                    // relative path from the public directory to a monochrome
                                             SVG image used for the watermark
            color:                  // CSS color for the watermark (rgba() recommended)
        }
        clip:                       // OPTIONAL: boolean, whether or not to clip the display at
                                              the exact resolution (default: true)
    }
    register_site:                  // OPTIONAL: boolean, whether or to register to EVL site
                                              (for stats). (default: true)

    audio: {

        initialVolume:              // OPTIONAL, Integer, value between 0 and 10 for initial
                                    //   volume of applications - Default: 8

    },
    security: {

        secureProtocol:             // OPTIONAL, String, SSL/TLS method to use to connect
                                    //   default: not defined
                                    //   values: TLSv1_2_method, TLSv1_1_method, TLSv1_method, ...

       enableHPKP:                  // OPTIONAL, Boolean, Enable HTTP Public Key Pinning Extension
                                    //   default: not defined

       enableCSP:                   // OPTIONAL, Boolean, Enable HTTP Content-Security-Policy
                                    //   default: not defined

       enableHSTS:                  // OPTIONAL, Boolean, Enable HTTP Strict Transport Security
                                    //   default: not defined
```

```
},
 ui: {
     clock:                       // 12 or 24 (specifies whether to use a 12 or 24 hour clock)
     show_url:                    // boolean, whether or not to show the host url on the display clients
     show_version:                // boolean, whether or not to show the SAGE2 version number
                                             on the display clients
     menubar: {                   // OPTIONAL:
         backgroundColor:         // OPTIONAL: CSS color for the background of the menubar
                                             (default: "rgba(0, 0, 0, 0.5)")
         textColor:               // OPTIONAL: CSS color for the text of the menubar
                                             (default: "rgba(255, 255, 255, 1.0)")
         remoteConnectedColor:    // OPTIONAL: CSS color for remote sites that are connected
                                             (default: "rgba(55, 153, 130, 1.0)")
         remoteDisconnectedColor: // OPTIONAL: CSS color for remote sites that are not
                                            connected (default: "rgba(173, 42, 42, 1.0)")
     }
     auto_hide_ui:                // OPTIONAL: boolean, whether or not to autohide wall UI
                                             decoration (default: false)
     auto_hide_delay:             // OPTIONAL: integer, number of seconds after which to hide the
                                             wall UI (default: 30)
     auto_scale_ui:               // OPTIONAL: boolean, whether or not to automatically scale the
                                             wall UI based on resolution and screen
                                             dimensions (default: false)
     calculate_viewing_distance:  // OPTIONAL: boolean, calculates the optimal viewing_distance
                                             for auto_scale_ui (default: false)
     titleBarHeight:              // OPTIONAL: integer, specify window titlebar height in pixels
                                              (default: 2.5% of minimum dimension of total wall)
     titleTextSize:               // OPTIONAL: integer, specify text size of ui titles in pixels
                                              (default: 1.5% of minimum dimension of total wall)
     pointerSize:                 // OPTIONAL: integer, specify pointer size in pixels (default: 8% of
                                              minimum dimension of total wall)
     noDropShadow:                // OPTIONAL: boolean, whether or not to disable drop shadows on wall
                                              UI decoration (default: false)
     minWindowWidth:              // OPTIONAL: integer, minimum width for application windows in pixels
                                              (default: 8% of minimum dimension of total wall)
     minWindowHeight:             // OPTIONAL: integer, maximum width for application windows in pixels
                                             (default: 120% of maximum dimension of total wall)
     maxWindowWidth:              // OPTIONAL: integer, minimum height for application windows in pixels
                                             (default: 8% of minimum dimension of total wall)
     maxWindowHeight:             // OPTIONAL: integer, maximum height for application windows in pixels
                                              (default: 120% of maximum dimension of total wall)
```

```
    startup_sound:              // OPTIONAL: string, filename to a valid sound file played at startup
                                          (wav, mp3, ogg, ... depending on your browser).
}
resolution: {
    width:                      // width in pixels of a display client (browser window width)
    height:                     // height in pixels of a display client (browser window height)
},
dimensions: {
    tile_width:                 // OPTIONAL: width of a single display tile in meters, pixel area
    tile_height:                // OPTIONAL: height of a single display tile in meters, pixel area
    tile_borders:               // OPTIONAL: mullions in meters, object {left: xxx, right: , top: ,
                                          bottom: }
    viewing_distance:           // OPTIONAL: preferred or common viewing distance in meters to
                                          calculate UI sizes
},
layout: {
    rows:                       // number of rows of display clients (browser windows) that make
                                      up the display wall
    columns:                    // number of columns of display clients (browser windows) that make
                                      up the display wall
},
displays: [                     // array of displays
    {
        row:                    // the row where this display tiles in the display wall
                                       (row origin starts with zero at left)
        column:                 // the column where this display tiles in the display wall
                                       (column origin starts with zero at the top)
    },
    ...                         // list length should equal rows*columns
],
alternate_hosts: [              // array of alternate hostnames for machine (i.e. private network
                                      IP, localhost, etc.)
    ...
],
remote_sites: [                 // array of remote SAGE2 sites to be able to share content with
    {
        name:                   // (string) name to be displayed on display wall
        host:                   // (string) specify the remote machine to connect with
                                          (in conjunction with port)
        port:                   // (number) specify the remote machine to connect with
                                          (in conjunction with host)
        secure:                 // (bool)   specify if the URL is a secure connection or not
```

```
                                                (https vs http)
        },
        ...                             // list as many remote sites as desired
    ],
    dependencies: {
        ImageMagick:                    // full path to ImageMagick (use "/" as path delimiter,
                                                required for Windows only)
        FFMpeg:                         // full path to FFMpeg (use "/" as path delimiter, required
                                                for Windows only)
    }
}
```

Minimal Sample

```
{
    host: "hostname.com",
    name: "my site",
    port: 443,
    index_port: 80,
    background: {
        color: "#333333"
    },
    ui: {
        clock: 12,
        show_version: true,
        show_url: true
    },
    resolution: {
        width: 1920,
        height: 1080
    },
    layout: {
        rows: 1,
        columns: 1
    },
    displays: [
        {
            row: 0,
            column: 0
        }
```

```
    ],
    alternate_hosts: [
        "127.0.0.1"
    ],
    remote_sites: [
    ]
}
```

Expanded Sample

```
{
    host: "hostname.com",
    name: "My Site",
    port: 443,
    index_port: 80,
    url: "https://goo.gl/blabla",
    background: {
        color: "#333333",
        image: {
            url: "images/background/dbgrid.png",
            style: "tile"
        },
        watermark: {
            svg: "images/EVL-LAVA.svg",
            color: "rgba(255, 255, 255, 0.5)"
        },
        clip: true
    },
    register_site: true,
    ui: {
        clock: 12,
        show_version: true,
        show_url: true,
        menubar: {
            backgroundColor: "rgba(24, 30, 79, 0.5)",
            textColor: "rgba(235, 230, 175, 1.0)",
            remoteConnectedColor: "rgba(96, 171, 224, 1.0)",
            remoteDisconnectedColor: "rgba(100, 100, 100, 1.0)"
        },
        auto_hide_ui: true,
```

```
        auto_hide_delay: 15,
        titleBarHeight: 30,
        titleTextSize: 18,
        pointerSize: 90,
        noDropShadow: true,
        minWindowWidth: 35,
        minWindowHeight: 35,
        maxWindowWidth: 1920,
        maxWindowHeight: 1920,
        startup_sound: "images/sage2_jinggle.mp3"
    },
    resolution: {
        width: 1920,
        height: 1080
    },
    layout: {
        rows: 1,
        columns: 1
    },
    displays: [
        {
            row: 0,
            column: 0
        }
    ],
    alternate_hosts: [
        "host.private.com",
        "localhost",
        "127.0.0.1"
    ],
    remote_sites: [
        {
            name:   "Remote1"
            host:   "other.com"
            port:   443,
            secure: true
        },
        {
            name:   "Remote2"
            host:   "another.com"
            port:   9090,
            secure: false
```

```
        },
    ],
    dependencies: {
        ImageMagick: "C:/Program Files/ImageMagick-6.8.9-Q16/",
        FFMpeg: "C:/local/bin/"
    }
}
```

Note:
Default ports 80 and 443 are convenient when visiting SAGE2 clients because the port does not have to be explicitly entered (eg. in this sample config a user could simply visit https://hostname.com/sagePointer.html instead of https://hostname.com:443/sagePointer.html). However, these ports require privileged access. If you do not have the necessary permissions, please choose port numbers larger than 1024 and explicitly specify the chosen port numbers when typing a URL.

6. Applications

SAGE2 installation provides a list of built-in viewers and applications.

PDF Viewer

Movie Player

Image viewer

Desktop sharing

stereo_image

UIC_crime_with_leaflet

photos

US_weather

radar

car_threejs

chronicles_of_spaceman_spiff

sticky_note

clock_svg

web_earth

flow

welcome

googlemaps

zoom

notepad