Specvis Desktop ver 1.1.1

Author: Piotr Dzwiniel

Contact: piotr.dzwiniel@gmail.com

Last modification: December 27th, 2021

CONTENTS

1 About Specvis Desktop

Specvis Desktop is a free, open-source and academically verified software for visual field examination using static perimetry, which consists of displaying visual stimuli in various places on the screen in the form of points of different brightness, to which the participant of the examination responds when he or she sees them. As a result, Specvis Desktop obtain a visual field sensitivity map based on the basis of which we can conclude about its condition.

2 Requirements

In order to ensure the multiplatformity of the application (which runs on Windows, Linux, and Mac), it was written in the Java programming language. This means that the application needs Java Runtime Environment (JRE; platform for running Java applications) to be installed on your computer to run. That said you need specifally JRE in version 8.121 or above, but not 10 and above due to the fact, that Specvis Desktop uses JavaFX for which Oracle (a company that officially develops Java) stopped its support starting with JRE 10. Nevertheless, if you must use JRE 10 or above, you can find installing openjfx as a solution to your possible problem with running Specvis Desktop. You just have to point to the openjfx directly and add required modules when launching Specvis Desktop from the command prompt by typing for example:

```
java --module-path openjfx/lib --add-modules javafx.controls,javafx.fxml -jar Specvis.jar
```

But I strongly suggest to stick to the JRE 8.121 if possible. You can check whether your computer has an appropriate JRE version by typing `java -version` in the command prompt. You can download a specific JRE version here - http://www.oracle.com/technetwork/java/javase/downloads/index.html.

3 Download

You can download Specvis Desktop by clicking the following link:

https://github.com/piotrdzwiniel/Specvis/raw/master/latest_build/Specvis_v1_1_1/Specvis.zip

What will be downloaded is a *.zip archive. Unzip it in a desired location. The content of the unzipped folder should contain Specvis.jar file. You will use this file to launch Specvis Desktop application. Zip archive contains also the following elements:

Element	Type	Description
Resources	Folder	Contains application resources like images.
Results	Folder	Contains visual field examination results.
Settings	Folder	Contains Specvis Desktop settings saved by the User in a form of *.sset files.
License	File (*.txt)	Text of the GNU GPLv3 license.
patients	File (*.s)	Specvis Desktop “database” with patients information.
screenLuminanceScales	File (*.s)	Specvis Desktop luminance scales.

4 QUICK START

4.1 LAUNCH

In order to launch Specvis Desktop double click on the `Specvis.jar` file or tye `java -jar Specvis.jar` in the command prompt, remembering, that you have to include Specvis directory in the command you want to execute or while being in the appropriate directory from the level of the command prompt.

4.2 DESCRIPTION OF THE GUI

Here I provide a brief description of the Specvis Desktop graphical user interface (GUI) to help you take first steps on the road to perform your first visual field examination.

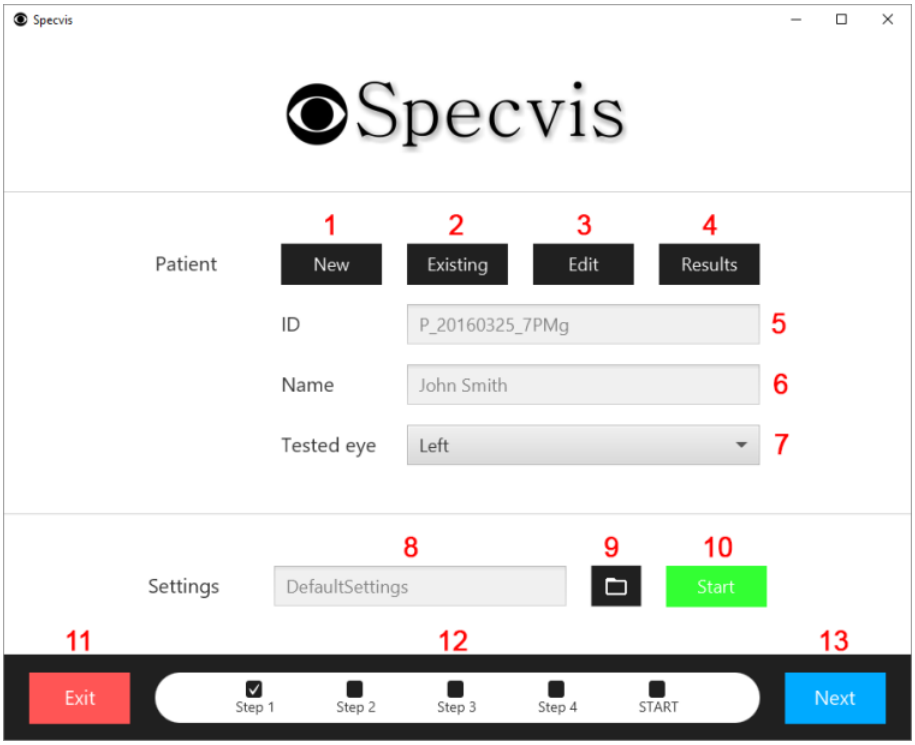


Figure 1. When you launch Specvis Desktop,this is the very first window which you will see. From its level you can manage patients and their results, choose the eye you want to examine, and choose previously saved or predefined settings. **1.** Button for opening window where new patient can be added to the database (Figure 2). **2.** Button for opening a window where an existing patient can be choosed from the database (Figure 3). **3.** Button for opening a window () where information about chosen patient can be changed (Figure 2 – editing patient information is performed in the similar window which is used for adding a new patient). **4.** Button for opening a with table containing visual field examination results of the chosen patient (Figure 4) – to use this function you must first choose a patient. **5.** Identificator of the chosen patient. **6.** Name of the chosen patient. **7.** Expandable list for choosing eye you want to test. **8.** Name of the chosen settings. **9.** Button for opening a window where you can choose previously saved settings. **10.** Button for moving directly to the procedure preview window (Figure 20) where you can start the visual field examination – you skip all settings windows. **11.** Button for exiting the application. **12.** Bar indicating step of the application configuration on the way to visual field examination – steps 2-4 contain various settings, which you can skip by clicking *Start*

button described before. **13.** Button for moving to next step window containing first set of the application settings, where you can change screen and luminance scale options if needed (Figure 8).

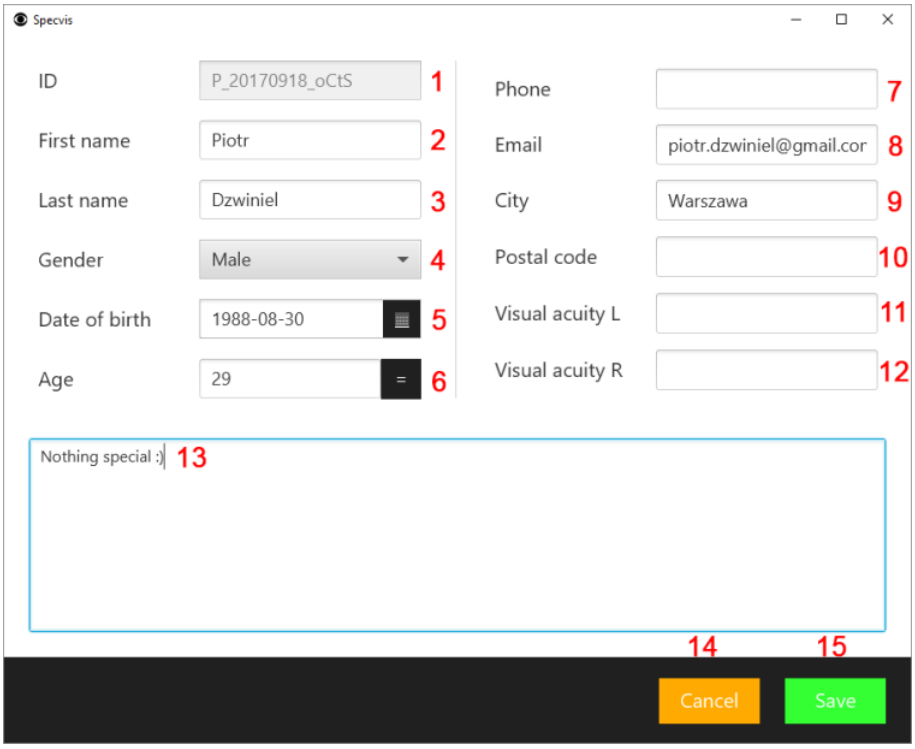


Figure 2. Window for adding a new patient to the database. **1.** Application first creates an unique identification number for each new patient you want to add. **2, 3.** Fields for providing first and last name of the patient. **4.** List for choosing gender of the patient. **5, 6.** Fields for providing information about patient's age. **7-12.** Fields for providing other information about the patient. **13.** Area for providing additional information about the patient which weren't included in the predefined fields. **14, 15.** Buttons for cancelling and saving the process of adding a new patient to the database, respectively.

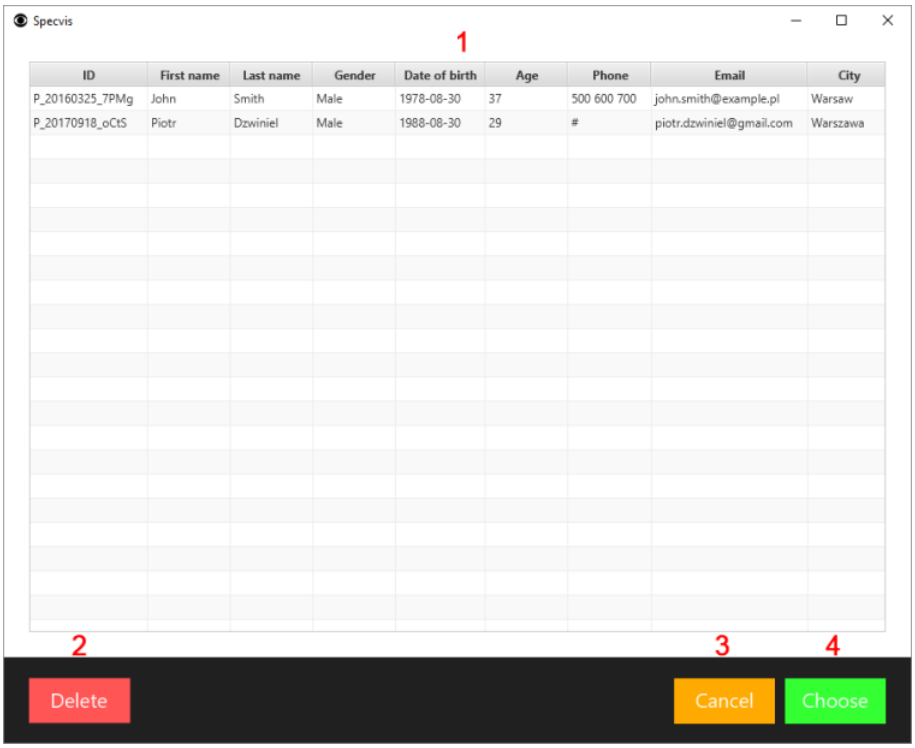


Figure 3. Window for choosing an existing patient from the database. **1.** Table view containing all existing patients in the database. **2.** Button for deleting a chosen patient. **3.** Button for closing the window. **4.** Button for choosing selected patient in the table.

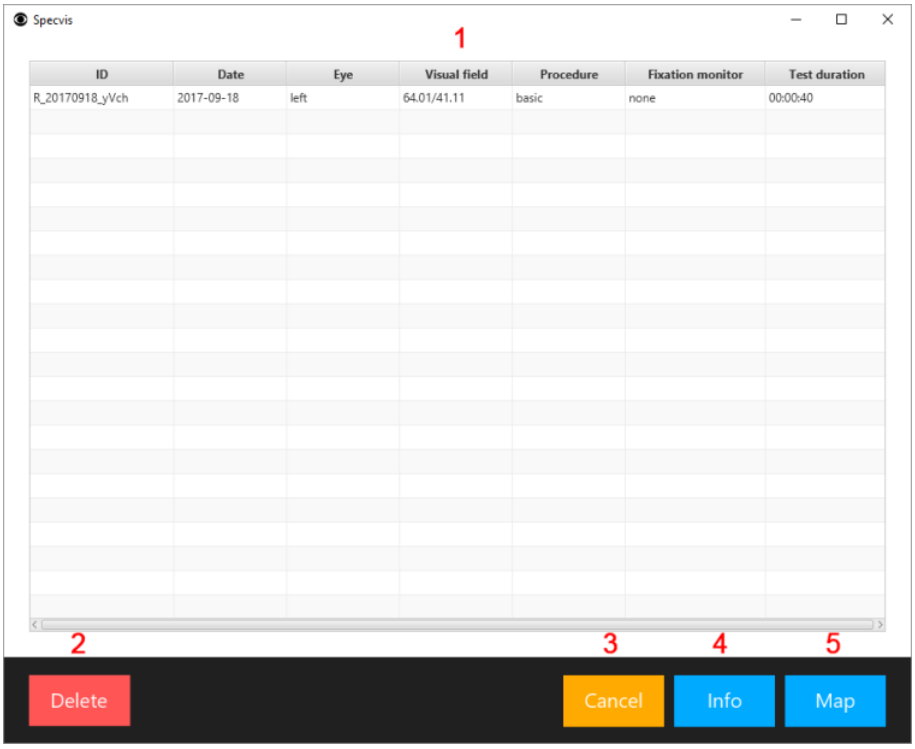


Figure 4. Window for previewing patient's results. **1.** Table containing all patient's results. **2.** Button for deleting chosen record. **3.** Button for closing the window. **4.** Button for opening the window with text information about the chosen record (Figure 5). **5.** Button for opening the window with graphical representation of the results in a form of visual field map (Figure 6).

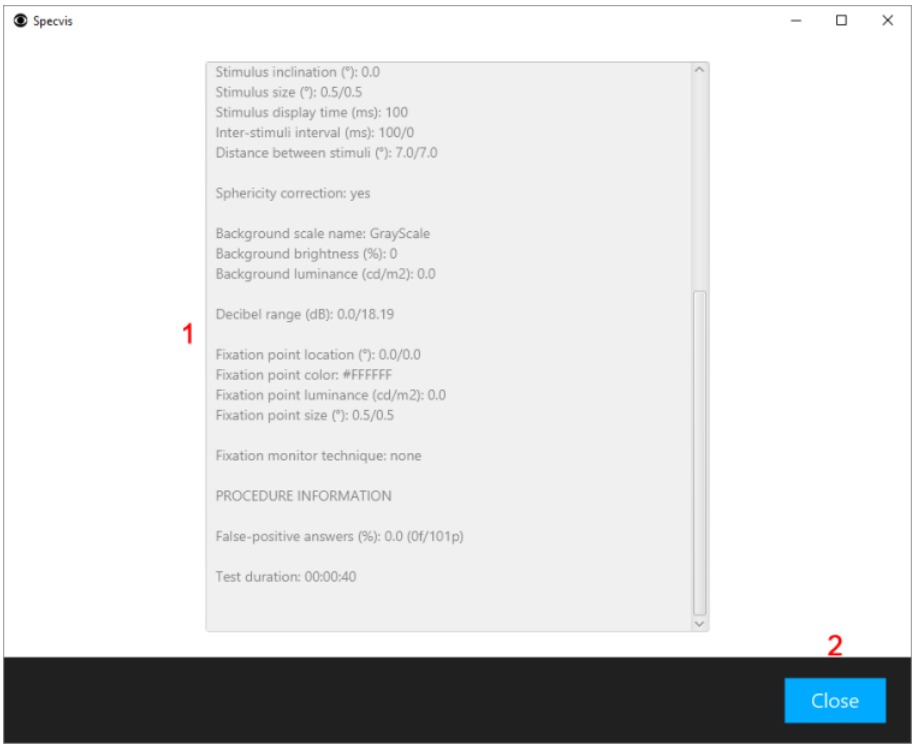


Figure 5. Window with text information about the chosen results record. **1.** Text area. **2.** Button for closing the window.

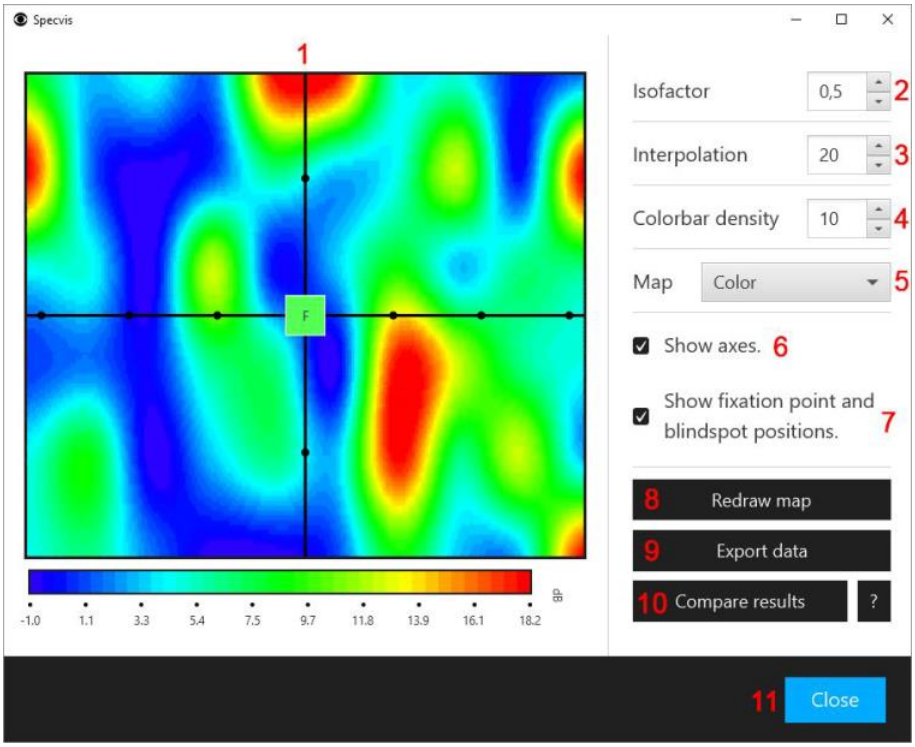


Figure 6. Window with graphical representation of the results in a form of visual field map. **1.** Visual field map with a colour scale expressed in decibels (dB). **2, 3.** Isofactor and interpolation values are used for changing "resolution" of the graphical map. **4.** Colour bar density value describes the number of ticks. **5.** List of available colour scales. **6, 7.** Checkboxes for including axes and fixation point / blind spot positions on the visual field map. **8.** Button for redrawing the map. **9.** Button for exporting the map as a two-dimensional, numerical matrix. It can be next used in other environments and

languages, such as Python or Matlab. **10.** Button for opening a window where you can compare current visual field map with the one that was acquired in the previous examinations. **11.** Button for closing the window.

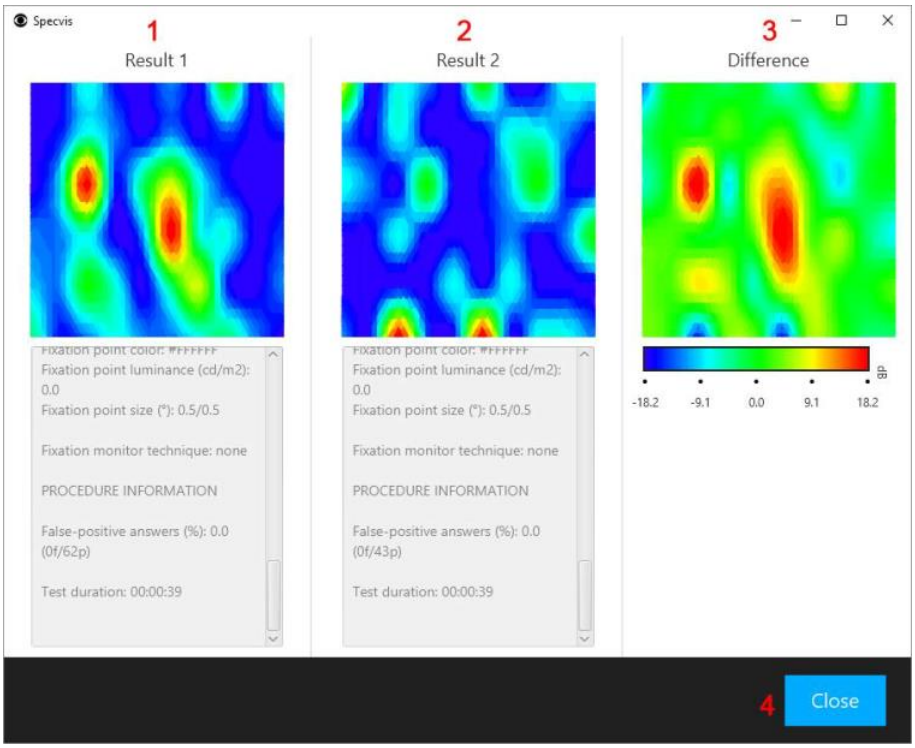


Figure 7. Window for comparing visual field maps. **1.** Visual field map and text information related to the examination results from the first compared record. **2.** Results for the second record. **3.** Graphical map illustrating difference between compared results, i.e. record nr 1 minus record nr 2. **4.** Button for closing the window.

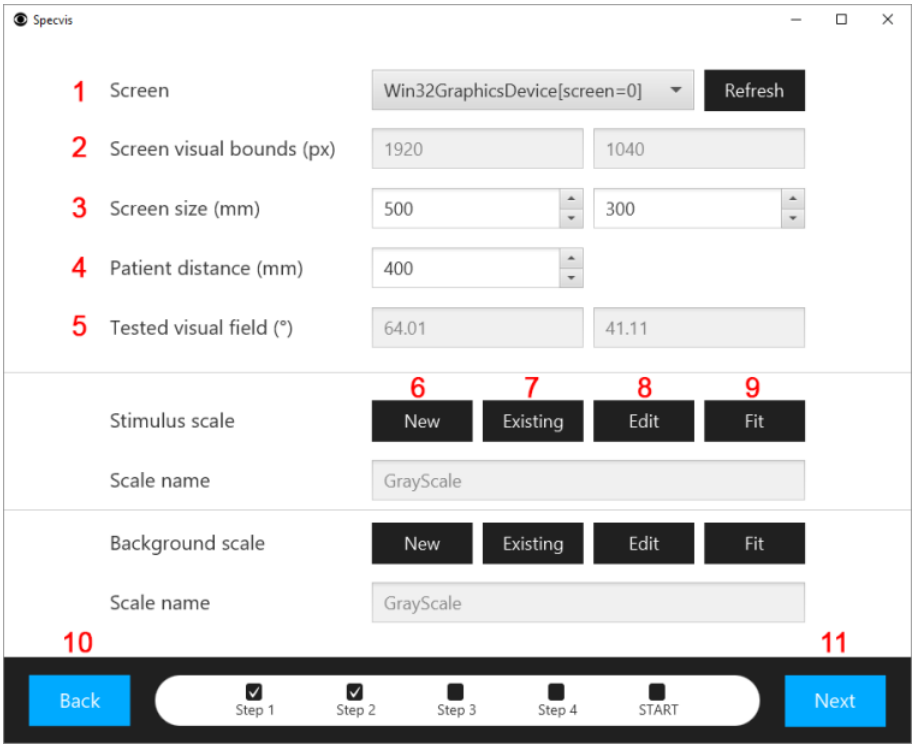


Figure 8. Window for setting screen and luminance scale options. **1.** List of the available screens (connected to the computer). Choose a screen which will be used to display the visual field examination procedure. **2.** Resolution of the selected screen. **3.** Define width and height of the chosen screen in mm. **4.** Define perpendicular distance between centre of the screen and patient's eyes in mm. **5.** Visual field that will be examined expressed in degrees. Calculation of the visual field is based on the size of the chosen screen and patient's distance from it, so make sure, that provided information are accurate. **6.** Button for opening a window for the creation of a new luminance scale (Figure 9). **7.** Button for opening a window with the list of existing luminance scales (Figure 10). **8.** Button for opening a window where you can edit information about the existing luminance scale (Figure 11). **9.** Button for opening a window with the information about how good in terms of statistical goodness of fit the chosen luminance is (Figure 11). **10.** Button for going back to the previous Specvis Desktop window (Figure 1). **11.** Button for going to the next Specvis Desktop window (Figure 12).

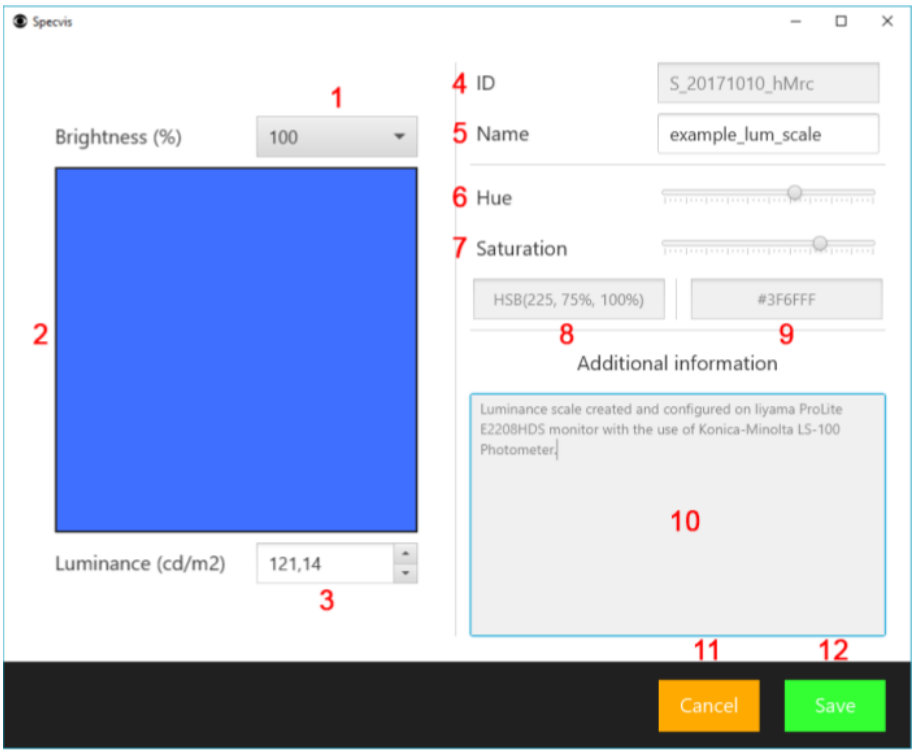


Figure 9. Window for the creation of a new luminance scale. **1.** Expandable list containing six subsequent brightness values, i.e. 0, 20, 40, 60, 80 and 100. **2.** Box for performing luminance measurement – after choosing brightness (**1**), hue (**6**) and saturation (**7**) set luminance meter to cd/m2, aim it at the centre of the box (**2**) and perform luminance measurement. The result write into luminance input field (**3**). **4.** Unique identification code generated for a new luminance scale. **5.** Name of the new luminance scale. **8, 9.** Fields containing HSB and HEX representation of the colour of the new luminance scale. **10.** Text area for providing additional information about a new luminance scale – for example provide here information about the luminance meter used in the process of the luminance scale creation or other. **11.** Button for closing the window and cancelling the process of the creation of a new luminance scale. **12.** Button for saving the newly created luminance scale.

IMPORTANT: If you want for your test results to be expressed accurately in cd/m^2 and comparable with other professional visual field testing equipment you should consider to create a new luminance scale which then will be used in your studies with the use of professional luminance meter. Remember, that created luminance scale is accurate for a specific screen you are using and its current

display settings. Change of the screen, its display settings or even change in the light conditions in the exam room should be a factor to adjust already created luminance scale in order to preserve its accuracy in expressing visual field examination results in cd/m^2.

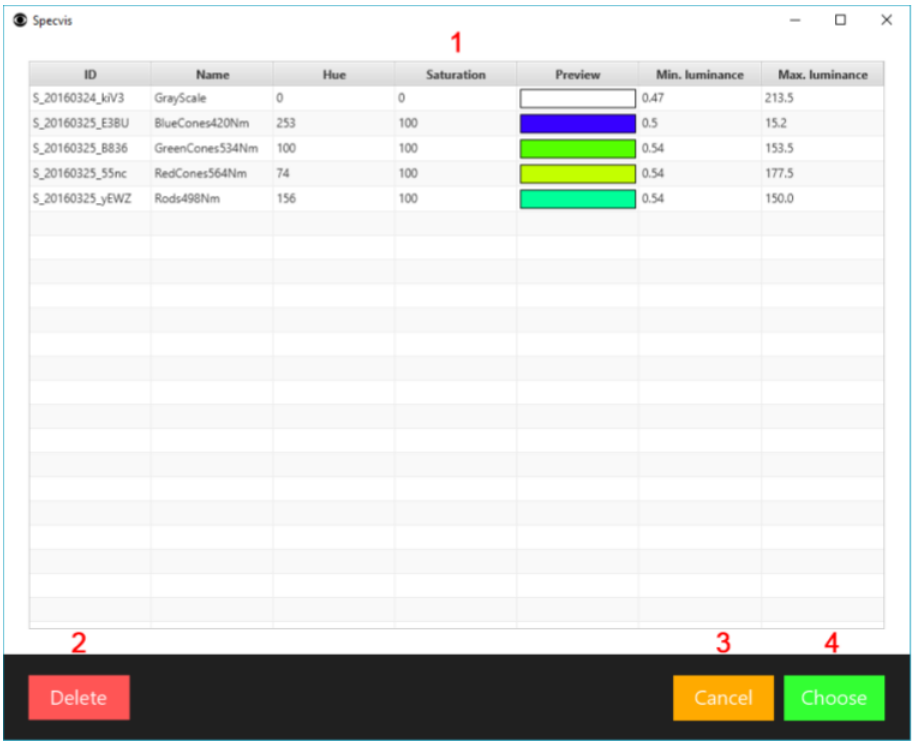


Figure 10. Window for choosing an existing luminance scale from the Specvis Desktop database. **1.** All existing luminance scales. **2.** Button for deleting a chosen luminance scale. **3.** Button for closing the window. **4.** Button for choosing selected luminance scale in the table.

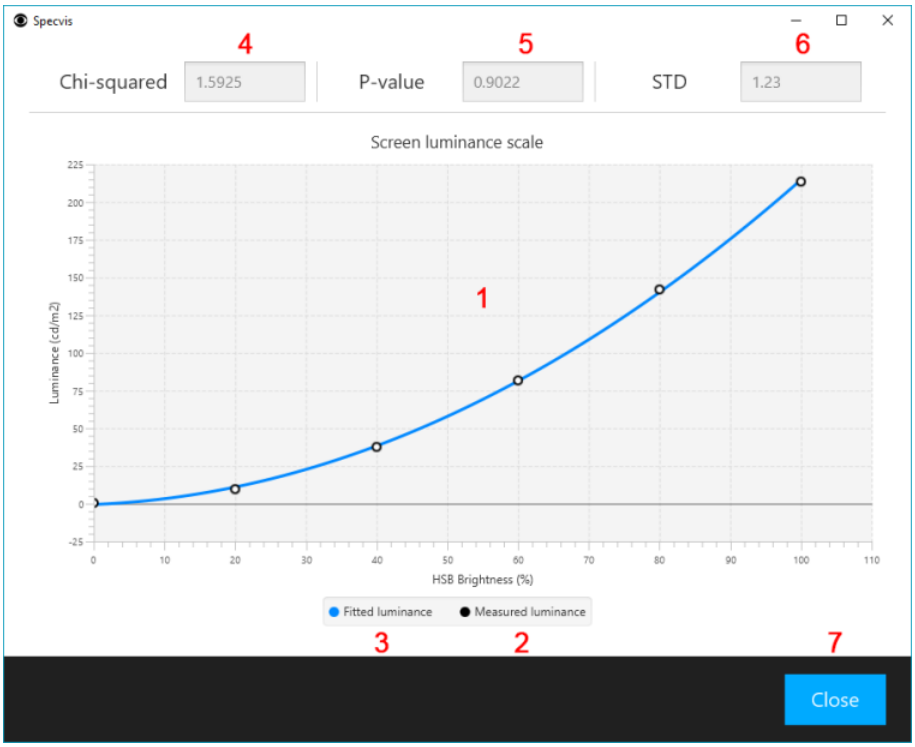


Figure 11. Window with the information about a statistical reliability of the created luminance scale. **1.** Chart illustrating measured luminance values with the use of luminance meter (**2**) and the fitted luminance (**3**) curve consisted of 101 artificially computer luminance values used by Specvis Desktop for pairing brightness values with luminance values. The application creates 101-element vector with luminance values based on only six real luminance measurements conducted during the creation of a new luminance scale. This vector provides information about luminance value of a given brightness value. For more information see section 'Software implementation' in *Specvis: Free and open-source software for visual field examination* article. **4, 5.** Chi-squared statistic and its significance for measured vs. fitted luminance values. **6.** Standard deviation for measured vs. fitted luminance values. **7.** Button for closing the window.

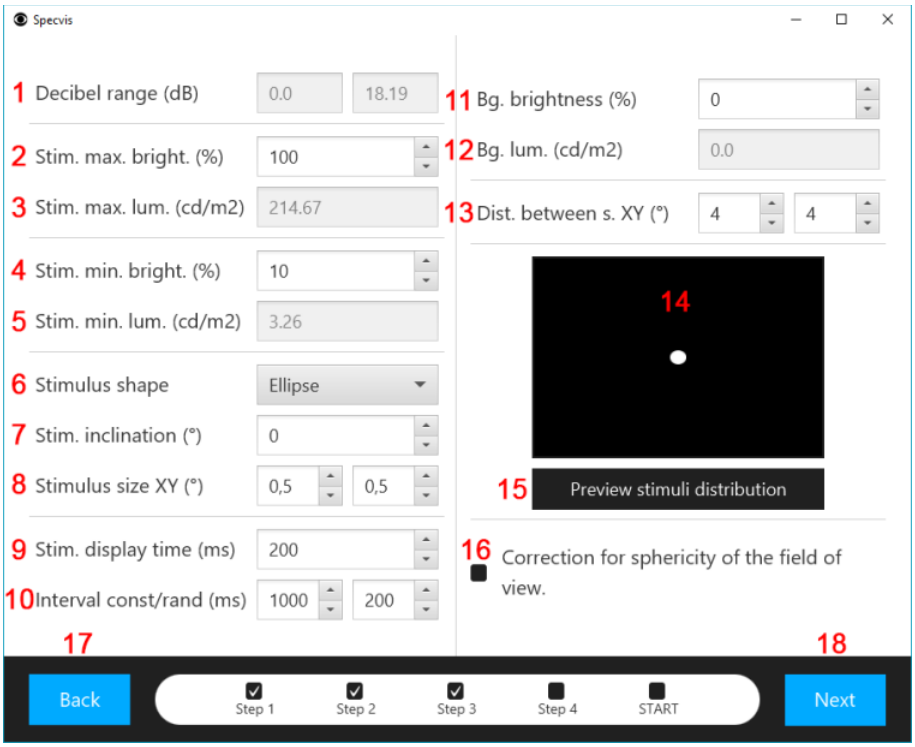


Figure 12. Window for setting visual stimulus and background options. **1.** Decibel range calculated for minimum and maximum stimulus luminance. **2-5.** Stimulus min/max brightness/luminance. Specvis Desktop assigns to a chosen brightness value a fitted luminance value. **6.** Stimulus shape. User can choose between ellipse and quadrangle (*Polygon*). **7.** Stimulus inclination in degrees. **8.** Stimulus width and height in degrees. **9.** Stimulus display time in milliseconds (ms). **10.** Time interval between last stimulus presentation and displaying next stimulus in milliseconds. Time interval consists of constant part and random part – for example, if constant part is equal to 1000 ms and random part is equal to 200 ms, time interval will vary between subsequent stimuli presentations in range between 1000 and 1200 ms. **11, 12.** Background brightness/luminance. **13.** Distance between stimuli locations in horizontal and vertical plane in degrees. **14.** Preview of the stimulus and background. **15.** Button for opening window with stimuli and fixation point locations (Figure 13). **16.** Checking this checkbox will result in turning on the correction for sphericity of the field of view for stimuli location distribution. **17.** Button for going to the previous Specvis Desktop window (Figure 8). **18.** Button for going to the next Specvis Desktop window (Figure 14).

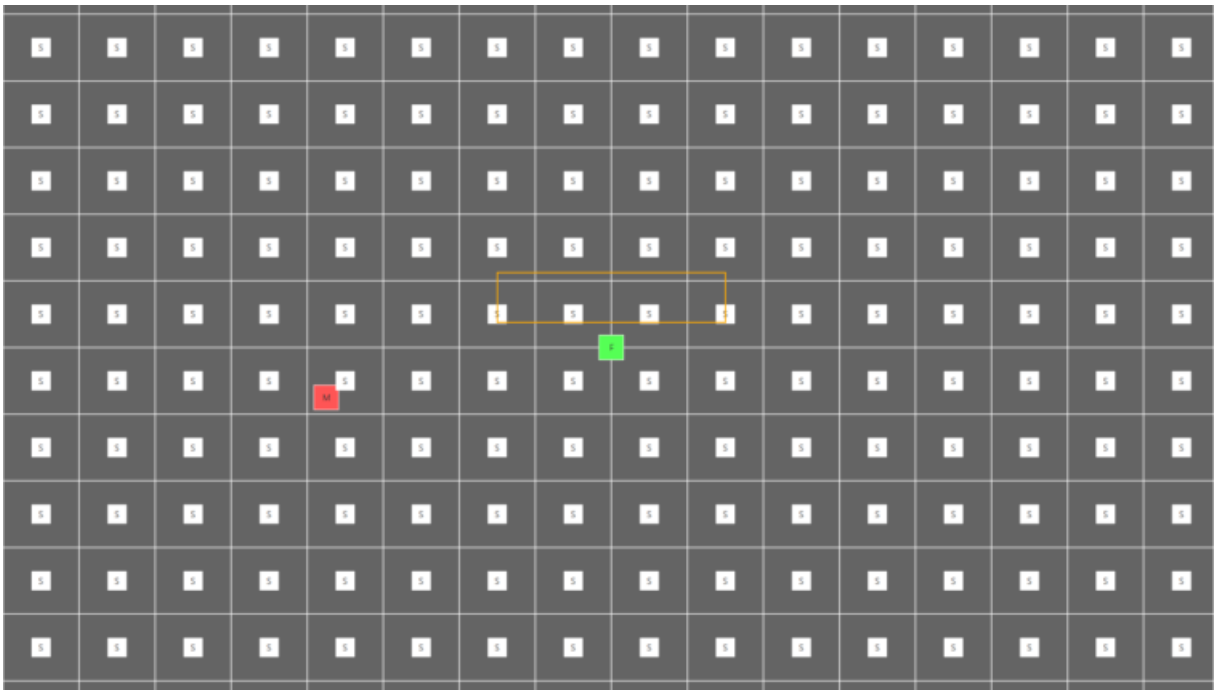


Figure 13. Window for previewing distribution of stimuli, blind spot and fixation point locations. S – stimulus location. M – location of the assumed blind spot location. F – fixation point location. Orange rectangle – location and area of the message box, which can be shown to the patient, when it loose its fixation.

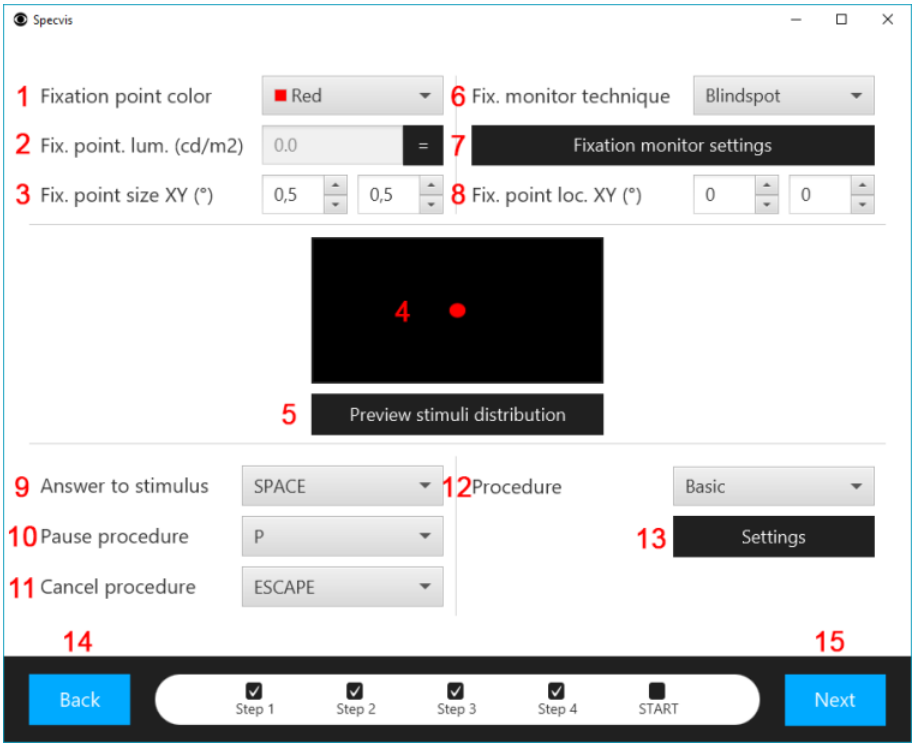


Figure 14. Window for setting fixation point, keyboard and visual field examination procedure options. **1.** Fixation point colour. **2.** Fixation point luminance (Figure 15). **3.** Fixation point width and height in degrees. **4.** Fixation point and background preview. **5.** Button for opening a window with stimuli, blind spot and fixation point locations (Figure 13). **6.** List with all available fixation monitor techniques. For more information about fixation monitor techniques see 'Software implementation' section in *Specvis:*

Free and open-source software for visual field examination article. **7.** Button for opening a window with settings dedicated for chosen fixation monitor technique. **8.** Fixation point location in degrees in a horizontal (left field) and vertical (right field) plane, respectively. **9-11.** Configuration of the keys used during the visual field examination procedure. **12.** List of available procedure algorithms. **13.** Button for opening a window with settings for a chosen procedure algorithm (Figure 19). **14.** Button for going to the previous Specvis Desktop window (Figure 12). **15.** Button for going to the next Specvis Desktop window (Figure 20).

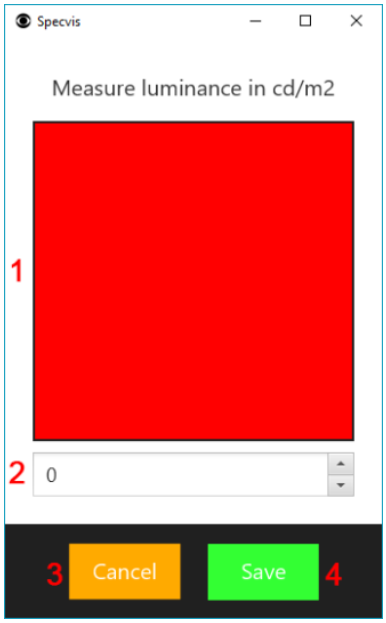


Figure 15. Window dedicated for performing a luminance measurement for the fixation point used in the visual field examination procedure. **1.** Using luminance meter set to cd/m2 point it at the centre of the box and perform luminance measurement. **2.** Input measured value into a field and hit *ENTER*. **3.** Button for cancelling the luminance measurement procedure and closing the window. **4.** Button for saving the luminance measurement and closing the window.

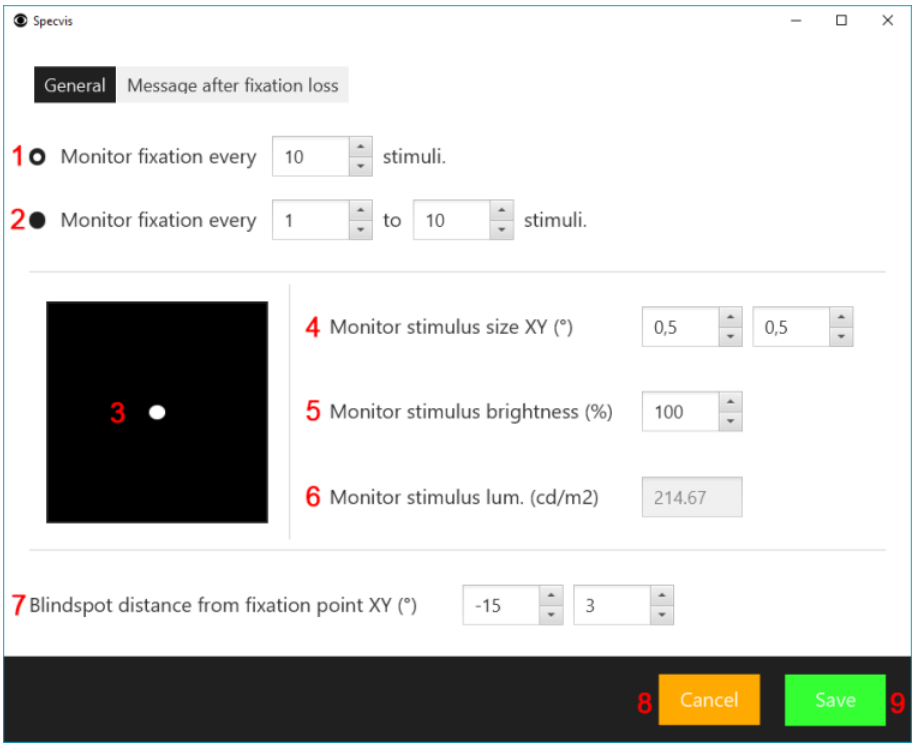


Figure 16. Window with settings dedicated for fixation monitor technique based on displaying a *control* stimulus in the predefined blind spot location. **1, 2.** Decide how often (in relation to visual stimuli presentation) the application should present a *control* stimulus in the predefined blind spot location monitoring the accuracy of the patien's fixation (gaze) during the test. **3.** Preview of the *control* stimulus and the background. **4-6.** *Control* stimulus parameters. *Control* stimulus use the same luminance scale as *ordinary* visual stimuli. **7.** Predefined blind spot location expressed as distance (in degrees in horizontal and vertical planes) from the fixation point location. For example, values X = -15 and Y = 3 (as shown on the figure) set predefined blind spot location 15° to the right and 3° above predefined point location. If X and Y equals 0, then predefined blind spot location is the same as the predefined fixation point location. **8, 9.** Cancel or save settings, respectively.

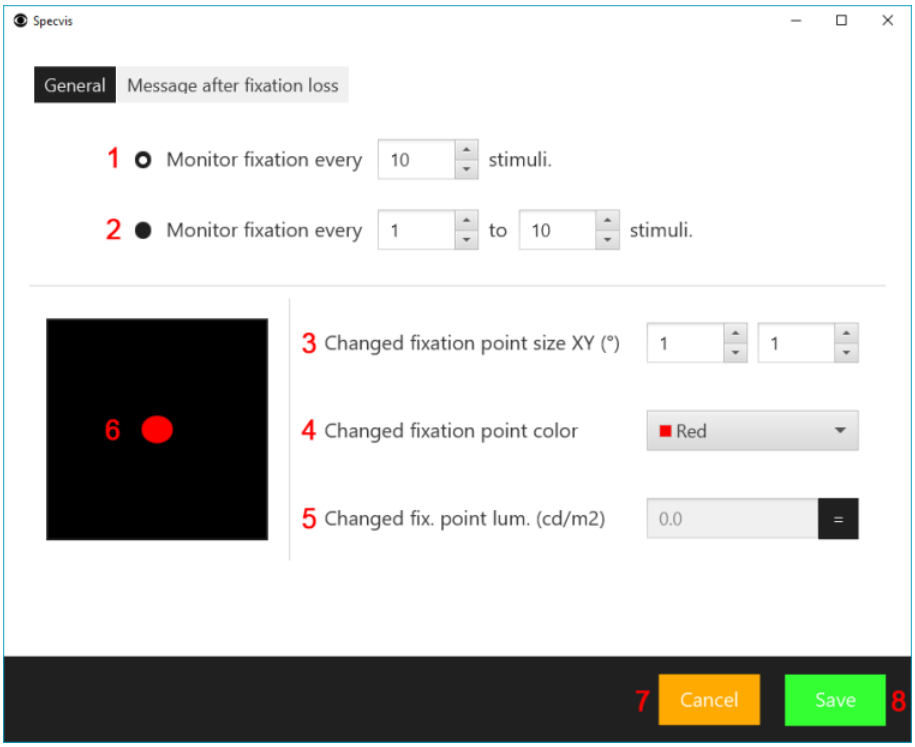


Figure 17. Window with settings dedicated for fixation monitor technique based on the changes of the fixation point itself. **1, 2.** Decide how often (in relation to visual stimuli presentation) Specvis Desktop should present *control* stimulus in the fixation point location. In *fixation point change* fixation monitor technique a *control* stimulus is a simple change of the fixation point characteristics such as size and colour. **3-5.** *Control* stimulus parameters. Similarly to the fixation point *control* stimulus is independent of the ordinary stimuli and background luminance scales. Thus, in order to assess its luminance, it's necessary to perform luminance measurement for it (**5**) (Figure 15). **6.** Preview of the *control* stimulus. **7, 8.** Cancel or save, respectively.

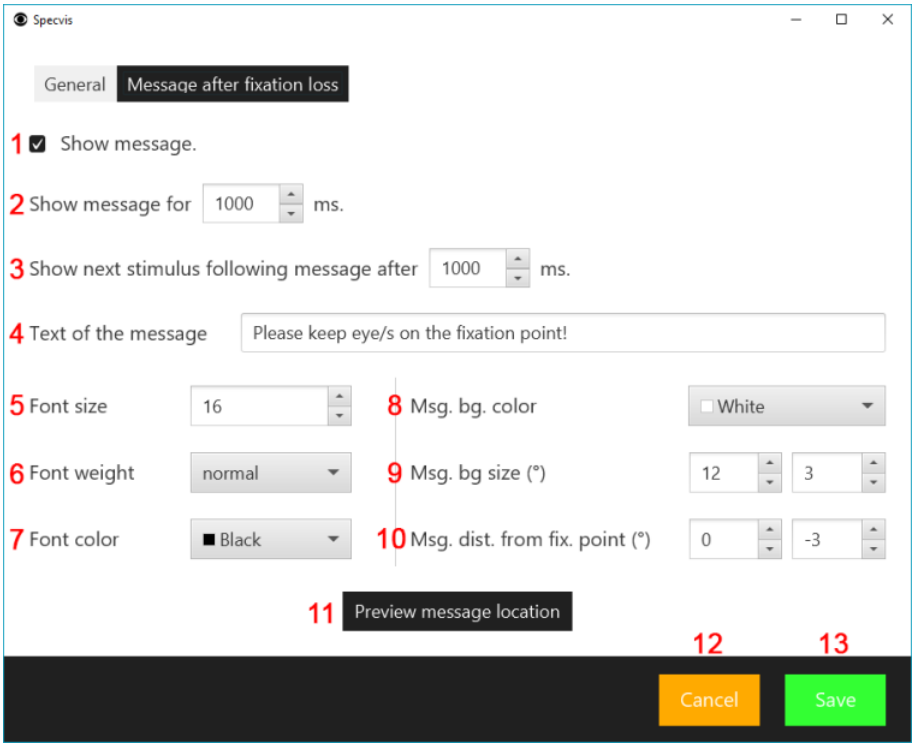


Figure 18. Window with settings common for all fixation monitor techniques. **1.** Show a specific message to the patient when he or she looses his or her fixation (gaze) understood here as a lack of the response to the *control* stimulus. **2.** Message display time. **3.** Display next ordinary stimulus after specific time interval. **4.** Text of the message. **5-9.** Parameters of the message's text and background. **10.** Location of the message box in relation to the fixation point location. **11.** Button for opening a window with stimuli, blind spot, fixation point and message box locations (**Figure 13**). **12, 13.** Cancel or save settings, respectively.

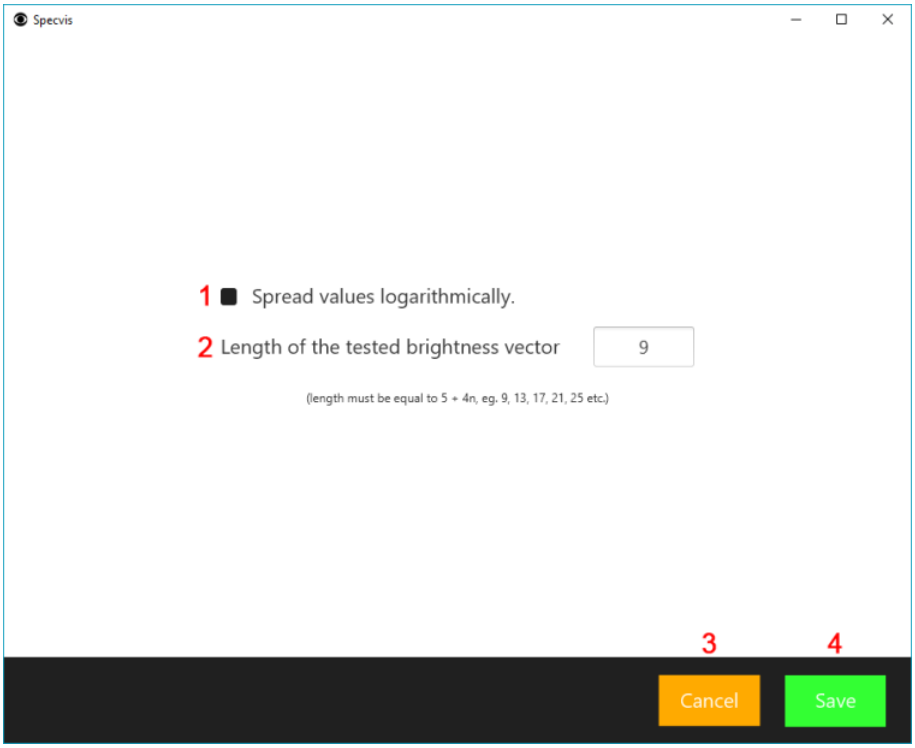


Figure 19. Window with the visual field procedure basic settings. **1.** Check if you want brightness vector values to be spread logarithmically. **2.** Length of the tested brightness vector. For deeper understanding of information provided in points **1** and **2** please see section 'Software implementation' in *Specvis: Free and open-source software for visual field examination* article. **3, 4.** Cancer or save settings, respectively.

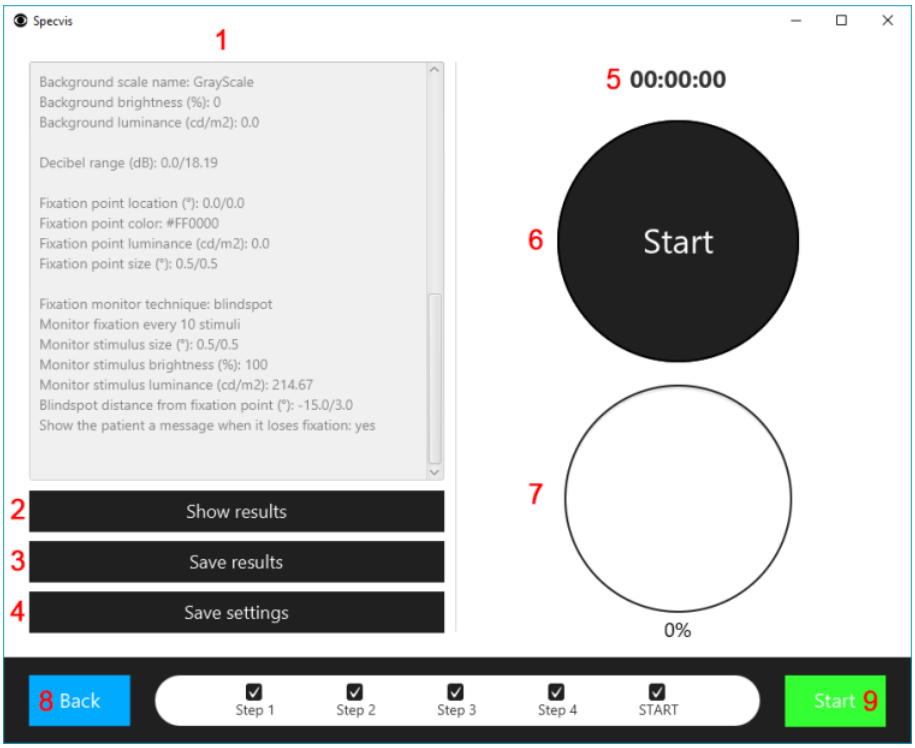


Figure 20. Window where you can start the procedure and observe by basic means its progress. **1.** Information about procedure settings. **2.** Button for opening window (Figure 6) with visual field examination results in a form of graphical map. **3.** Button for saving visual field examination results. When clicked, application creates in *Results/patient_location* an individual folder for storing results from this particular visual field examination procedure (for each examination application creates an individual folder). Specvis Desktop saves two files – *session_data.txt* and *session_info.txt*. First one stores information about locations of the visual stimuli used in the examination and brightness/ luminance/ decibel threshold values below which particular visual stimuli were not perceived by the patient. Columns in this file are as follows: 1) number of the stimulus; 2-3) stimulus location in pixels in horizontal and vertical planes – coordinates X = 0 and Y = 0 stands for top-left corner of the screen; 4-5) stimulus distance in degrees from the fixation point location – distance X = -3 and Y = -3 means, that stimulus location was 3° to the left and 3° above the predefined fixation point location; 6-8) brightness/ luminance/ decibel threshold value, below which patient did not perceive the stimulus. The second file stores information from the area described in (**1**). **4.** Button for saving procedure's settings used for the examination. Settings are saved in *Settings* folder. **5.** Procedure timer. **6.** Procedure state indicator. **7.** Procedure progress indicator. **8.** Button for going to the previous Specvis Desktop window. **9.** Button for starting the procedure.

4.3 Starting New Visual Field Examination Procedure – Diagram

Specvis Desktop, despite of its relatively wide configuration capabilities, is very easy to start working with. After downloading the application you have to walk only trough 5 simple steps in order to start basic visual field examination (Figure 21).

1. Launch Specvis Desktop by double clicking on Specvis.jar or typing in the command prompt java -jar Specvis.jar.
2. Choose patient by adding a new one or choosing an existing one from the application's database.
3. Choose the eye you want to test (left, right or both).
4. Choose settings you want to use.
5. Start the visual field examination.

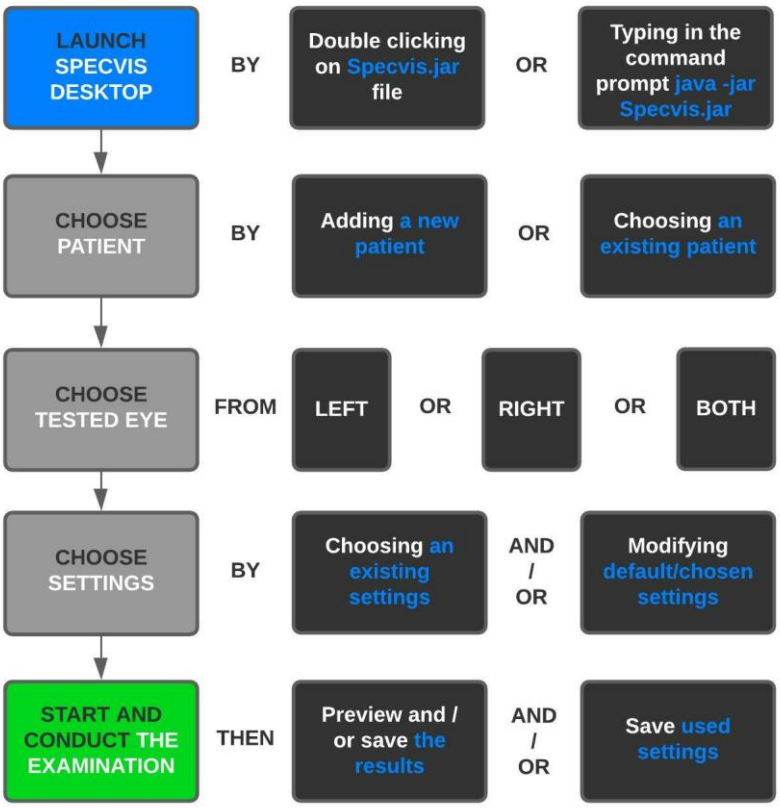


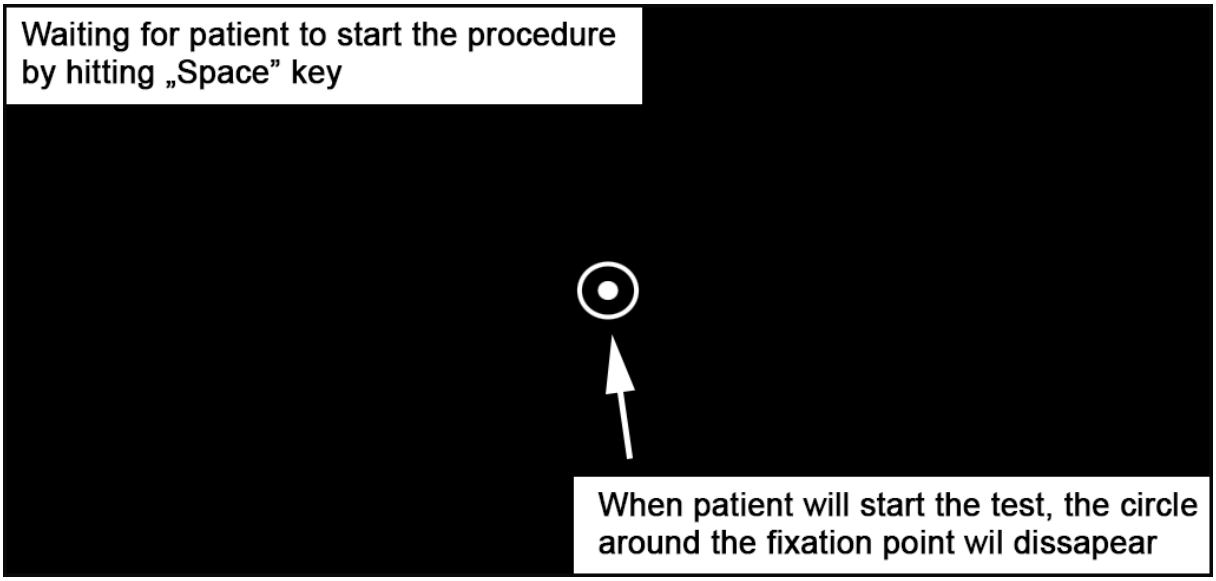
Figure 21. Specvis Desktop basic workflow can be stripped to five main steps consisting of launching the application, choosing the patient, choosing the patient's eye you want to test, choosing the settings you want to use for the examination, and performing the visual field test itself.

4.4 Description of The Visual Field Examination Procedure

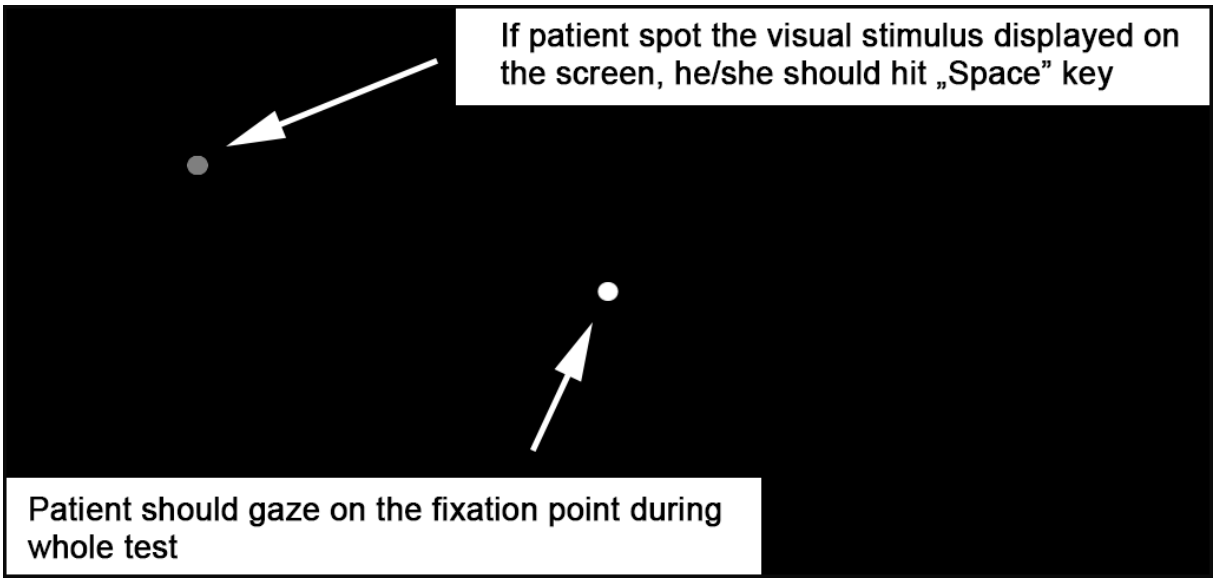
From the patient point of view you have to have in mind only three things during the test:

- Gaze (fixate) at the central point during the whole test.
- Hit *Answer to stimulus* key (in default its *Space* key) on the keyboard whenever you spot a dot displayed somewhere around the gazing (fixation) point on the screen.
- Hit *Answer to stimulus* key on the keyboard whenever gaze (fixation) point will change its colour or shape.

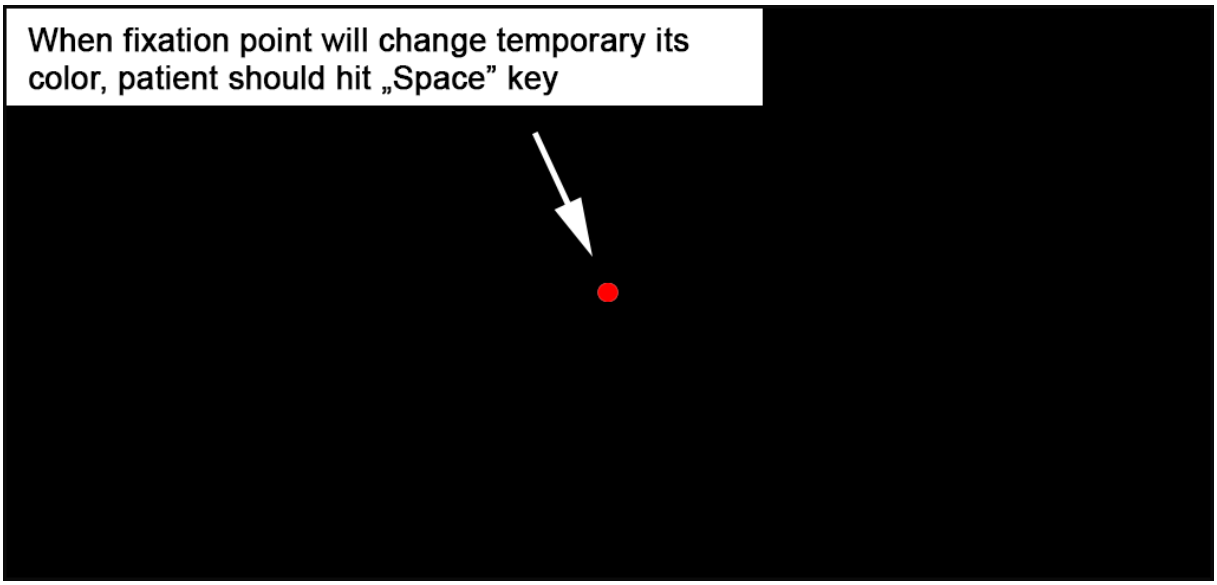
Knowing this, when visual field test will be started by the person responsible for conducting the test, you will see on the screen something like this (without descriptive text boxes of course)…



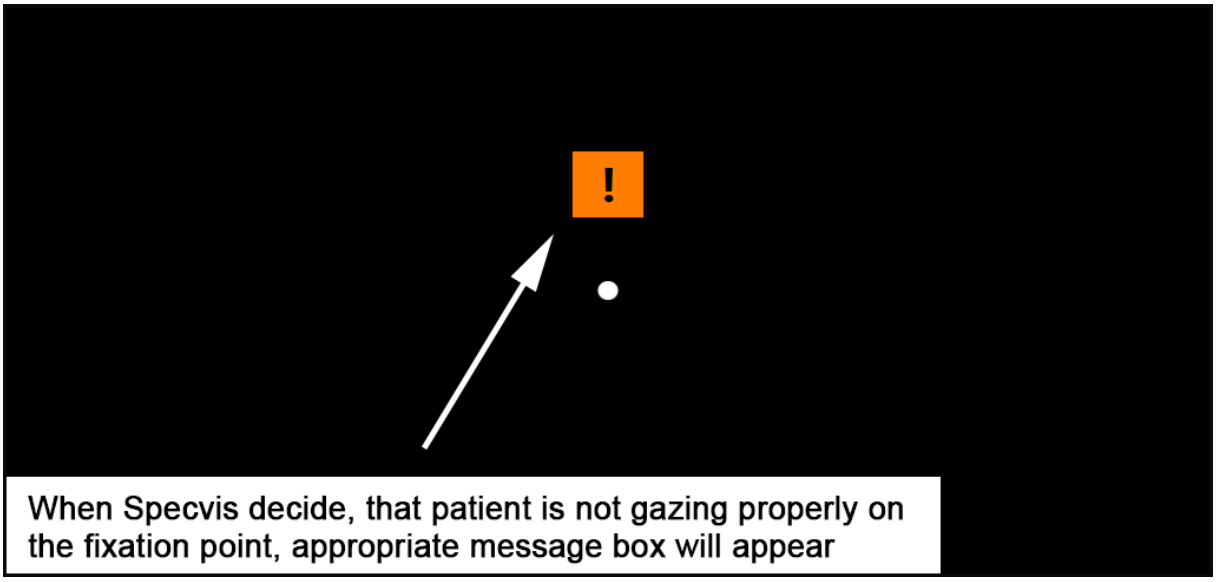
As a patient, you decide when the test will start by simply hitting *Answer to stimulus* key. When procedure is running visual stimuli with varying brightness will be displayed in random locations around the gazing (fixation) point on the whole screen. Your task, as a patient, is to respond to them by hitting *Answer to stimulus* key while gazing the whole time on the gazing (fixation) point…



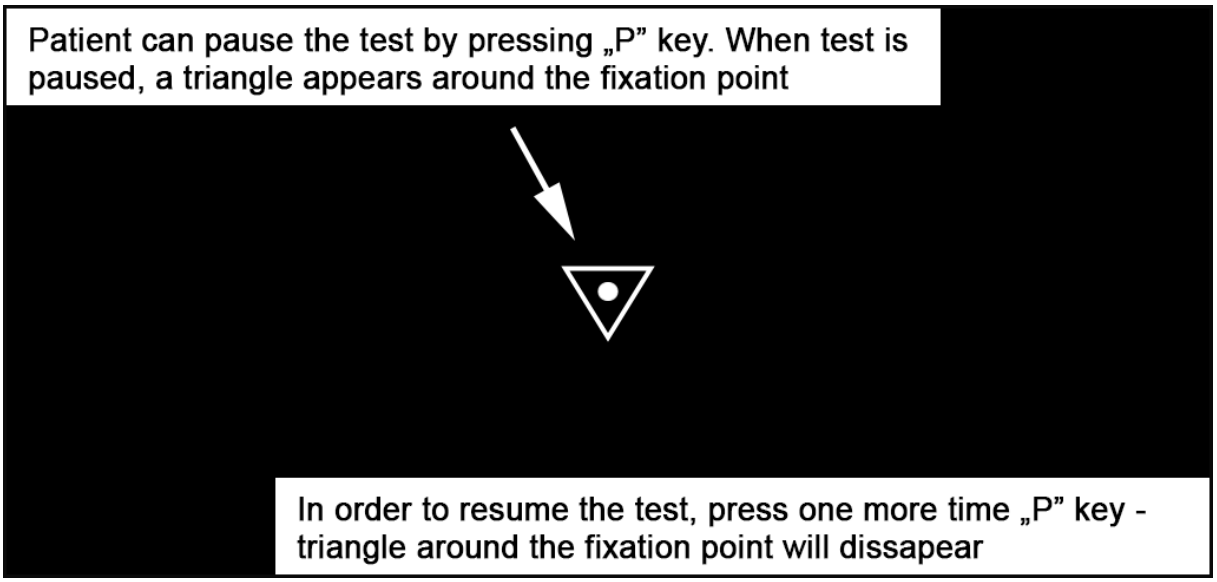
You should also hit *Answer to stimulus* key when gazing (fixation) point will change temporary its colour and/or shape…



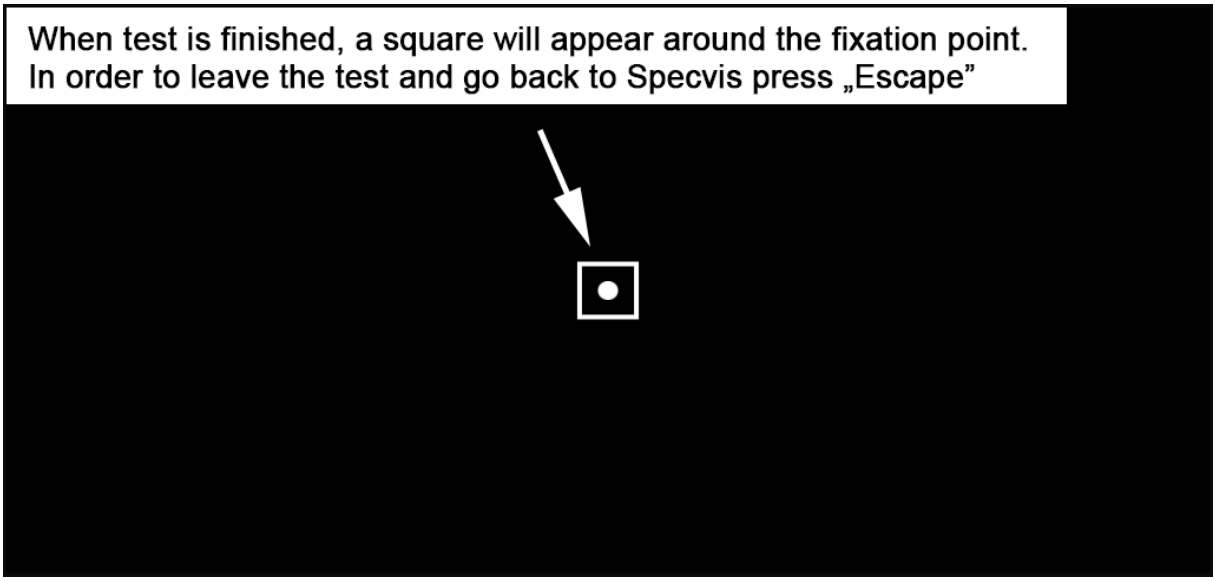
When the application will suspect that you are not gazing (fixating) on the gazing (fixation) point properly you will see an exclamation mark in an orange box above the gazing (fixation) point in order to remind you that you should gaze (fixate) properly. Wait for the exclamation mark to disappear by its own…



Furthermore, in any time of the test you can pause the test by hitting *Pause procedure* key (in default its *P* key) on your keyboard and resume it the same way in any given time. When the procedure is paused appropriately a triangle will appear around the gazing (fixation) point. When the procedure is resumed the triangle will disappear…



When your test is finished a square will appear around the gazing (fixation) point. You can now press *Cancel procedure* key (in default its *ESCAPE* key) on your keyboard to go back to the application window and for example preview/save the test results…



IMPORTANT: Procedure can differ based on the chosen settings. However, described rules are universal for all tests.

4.5 Previewing the Results

To preview the results from your test just click *Show results* in the window where you started the test (Figure 20). Window with the graphical representation of the results will be opened (Figure 6). In order to make your results look "better" try to change *Isofactor* to 0.5 and *Interpolation* to 10. You can also consider to change *Map* from *Monochromatic* to *Color*. After all changes click *Redraw map* to redraw the map. Don't worry, if the application will "freeze" for a second – the lower the isofactor and the higher the interpolation values the more computation has to be done by the application.

5 LICENSE

Specvis Desktop is currently licensed under GNU GPLv3but it can change for its future, upcoming versions. However, future versions of Specvis Desktop will always include the basic functionality to reliably evaluate the field of view for free. Accesibility of the application was my main idea when creating it, so I can asure you, that it will not change in the future.

6 SUPPORT

If there is any technical problem with Specvis Desktop, please go to the issues section on:

https://github.com/piotrdzwiniel/Specvis/issues

While there try to look for the solution to your problem among existing topics. If you can't find the solution, than create a new issue and describe your problem as accurate as it is possible. If something is not working in Specvis Desktop (it freezes, behave oddly etc.), then try to run it from the command prompt and then try to force this situation when something is not working. If any error will appear in the command prompt, copy it and include it in the description of the issue. Remember, the more information you'll provide about the problem you've encountered using Specvis Desktop, the bigger chance that this problem will be solved and will not occur any more.

7 SOURCE CODE

Specvis Desktop complete source code is publicly available on GitHub at the following link:

https://github.com/piotrdzwiniel/Specvis-Desktop

8 CONTRIBUTE TO SPECVIS DESKTOP DEVELOPMENT

Specvis Desktop was created and maintained by one person, but is meant to be supported by the community of academics, clinicians and casual users. Support can be understood as a contribution to the application's source code, but also as a share of ideas that can improve Specvis Desktop application. If you want to support this project by coding please read *TODO List for The Next Application's Version* and *Coding Guidelines*. If you want to support this project by sharing your ideas please read *TODO List for The Next Application's Version* and write to me directly at piotr.dzwiniel@gmail.com describing your idea.

8.1 TODO LIST FOR THE NEXT APPLICATION VERSION

- [functionality] Results from the visual field examination downloadable in a form of *.pdf printout similar to those in the professional equipment. Printout should contain all patient information and visual field map presented in a clinically standardized way.
- [functionality] Inclusion of reaction times to the visual stimuli during visual field examination.
- [gui] Reorganization Specvis Desktop graphical user interface into a "traditional" one window, menu bar and possibility to preview the ongoing visual field test.
- [dependencies] Specvis Desktop requires currently specifically JRE in version 8_121 to work appropriately which raise multiple problems for the end user.

- [other] Inclusion of the healthy visual field maps for a specific age and gender so their can be used as a baseline for the verification of possible visual field deficits for a specific patient.

8.2 Coding Guidelines

8.2.1 Before You Start to Write Any Code

Before start coding please make sure, that you are acquainted with *TODO List for The Next Application's Version*. However, I strongly support proposing ideas, that are not included in the aforementioned list – you can have a cool idea that I never thought about. Next, please open a new *Issue* on https://github.com/piotrdzwiniel/Specvis-Desktop/issues, describe what you want to change in/add to code (and why / how it will improve the application) so we can discuss it and approve it publicly – being on the same page before any coding is done can save a lot of time for everyone involved in the process.

Any contributions to the code development should be done via *pull requests* on GitHub. I assume that you are acquainted with contributing to GitHub projects and you have already a GitHub account. If yes, the following guide will help you to start your contribution to the code development with the use of IntelliJ IDEA environment (https://www.jetbrains.com/idea/) – if you work with other IDE, that's fine.

1. Download, install, and launch IntelliJ IDEA.
2. Log into your GitHub account via IntelliJ IDEA by choosing *File* → *Settings* → *Version Control* → *GitHub*. Provide all necessary personal information. Choose github.com as *Host*. Provide your GitHub's login and password credentials and click *Test* to test IntelliJ's connection with your GitHub's account. If everything is fine click *Apply* and/or *OK*.
3. Clone Specvis Desktop repository https://github.com/piotrdzwiniel/Specvis-Desktop.
4. Do not work on *master* branch. Work only on branch *version-X-X-X* (e.g. *version-1-2-0*).

8.2.2 Branches

Specvis Desktop repository will always have two active branches, i.e. *master* and *version-X-X-X*. The first one is considered as stable and is not devoted for the development process. Work only on the second one.

8.2.3 Pull Requests

Please make sure that each pull request has its corresponding opened issue with a thorough description of the code development. Please stick to the rules:

- Issue before pull request – Before starting any code development please create an issue for it and describe what you want to implement.
- One issue for one implementation – Try to enclose your code development into a easily descriptive implementations. Even, if one implementation will require few pull requests describe them in the same issue if they refer to the same implementational process.
- Content is equally important as its form – Try to avoid cosmetical changes in code, but at the same time remember, that content of your code/implementation is at the same level of importance with clarity and readability of your code/implementation.

8.2.4 Code Style

- Please follow Google Java Style Guide - https://google.github.io/styleguide/javaguide.html.
- Please write self-descriptive code with detailed javadoc comments.
- Please stick to the "clean code" rules - https://bit.ly/3JjCxVn.

9 CONTACT

For any questions write at piotr.dzwiniel@gmail.com.