

Design Document

Table of contents

Design Overview	3
Problem	3
System Organization	4
System Operation	5
System Architecture	6
System Design	6
Component Design	6
User Interface Design	7
Maintenance Scenarios	7

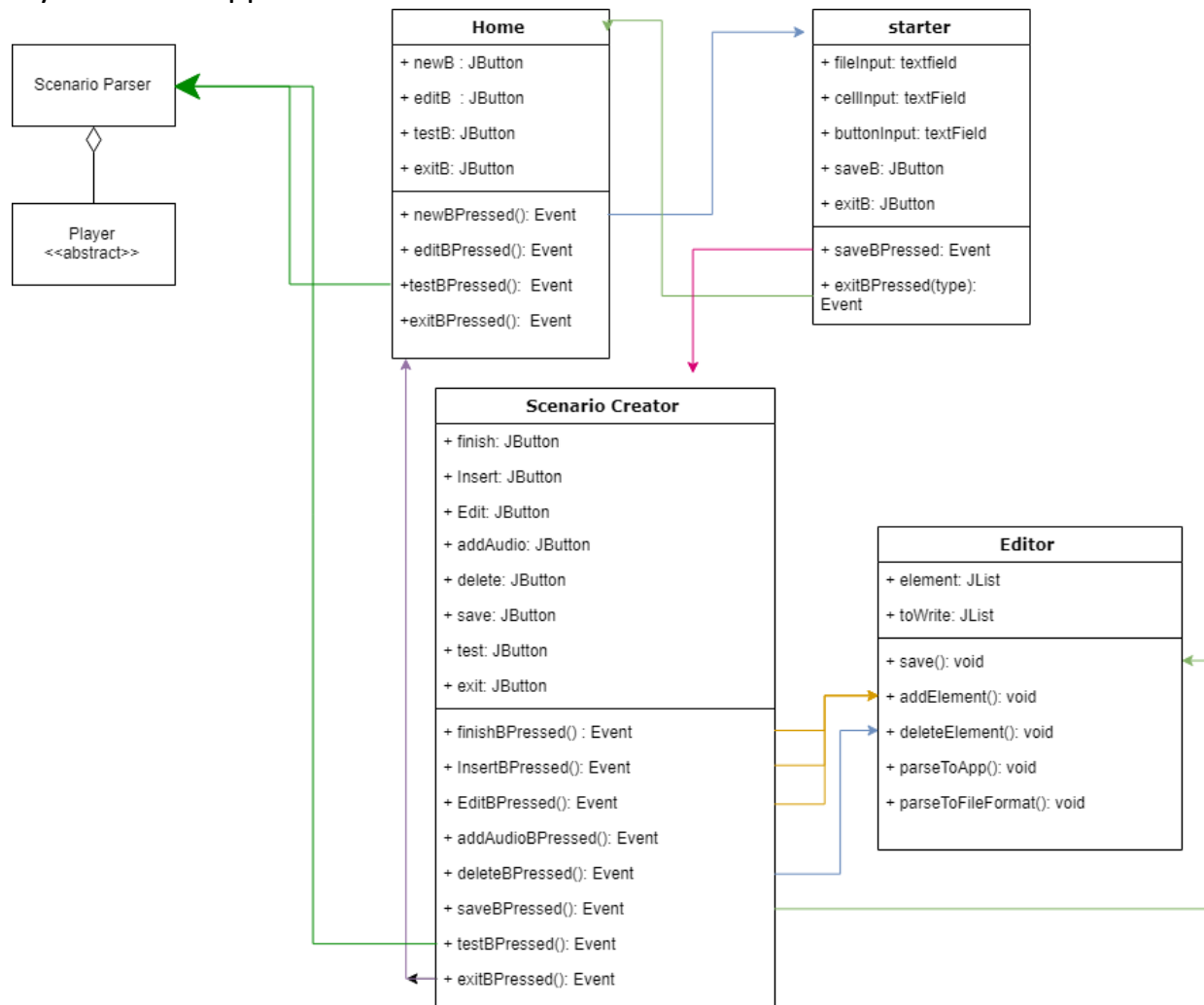
Design Overview

Problem

Since braille has become more popular to read amongst the visually impaired and their peers, family and friends, we were tasked with helping to come up with an efficient way to solve this. We were given knowledge of a device known as the Treasure braille box (TBB) which is a physical unit that can ask a user questions and set braille cells according to some data. For us we needed to create a way to make this data as easy and efficient as possible. To do this we were given a software simulation of the TBB and an overview on the different commands it could read. The objective was to come up with an application that takes all the information including the virtual TBB and information on how the commands are read, and make it into an application that allows users to create the scenario files for the TBB. We needed to make the application easy to use and understand while providing essential features such as creating, editing and testing the files. We were also given the task of making the application easy to operate for someone who may be visually impaired themselves.

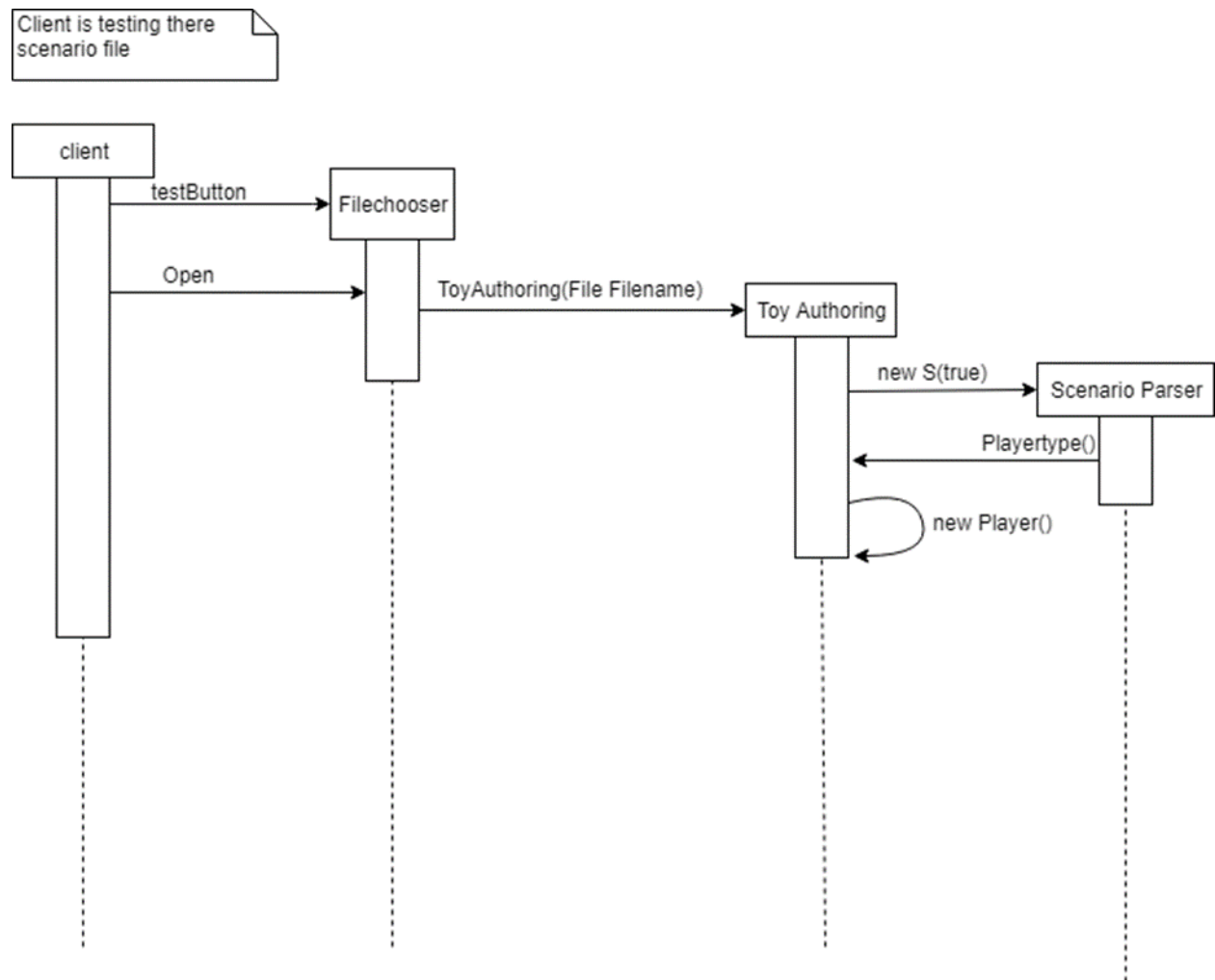
System Organization

For the system organization we needed to have the goal of a user either creating or opening a scenario file and being able to program it using easy to understand buttons and inputs. Below you can see the diagram of how we organized the layout of our application



System Operation

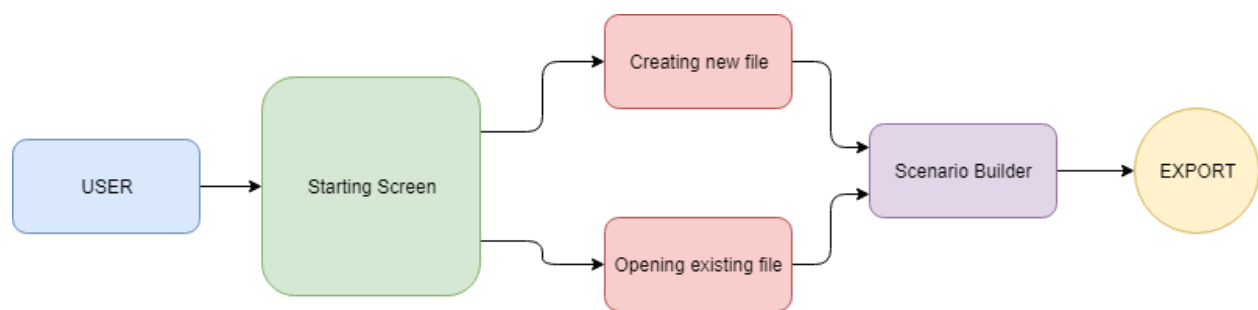
Below will be an example of the flow to test the scenario you have already created. The purpose is to show you the process of how the file is set to run from start to finish. This allows you to trouble shoot any issues if need be.



System Architecture

System Design

Below is a block diagram which shows the flow of the authoring app. It addresses it from a users perspective and shows the path the user would take when they launch the application.



Component Design

This section will explain what each part of the application does.

Start Menu

This part of the menu is where the user can choose what they want to do when opening the app. It gives them the option to create a new file, edit and existing, run a file or close the application. Clicking on new will prompt the user with a new menu which lets them set the name of the file as well as the buttons and cells they would like to include. Clicking edit prompts the user with a file chooser where they can select the file they want to open. Clicked on run opens a file chooser where the user can select the file they wish to run as well.

Scenario Builder

This part of the menu is used to design scenarios. In this area you will find a bunch of buttons and text prompts. Each button and text area is labeled for the

user to understand what goes there. They also all contain accessibility functions that help users with text readers for their computer. Clicking on these buttons do not open up and new windows but rather add information to the file or test the file.

User Interface Design

The concept behind this design was to keep things as simple as possible for the user while keeping all the core functionality. A simple color scheme was chosen that is not too bright and does not offer a variety of different colors. This should help people who may be color blind see the information better. The design itself also features large buttons and text areas to help users who may not have a steady hand or can't see the buttons have a better chance of clicking them. All elements also feature accessibility content which can be used to help navigate the menus. We decided not to include all the functionality that is available in the braille readers arsenal of applications, due to the fact that it could be confusing to the user. Should users want new options we can easily scale the display to add new options to better suit them. We still do pertain to the idea of keeping the options simple so the user can have more of a hassle free experience.

Maintenance Scenarios

If there is a need to rearrange the elements in the start screen, navigate to the starter.java class and under the initComponents() method you can change positions of the elements. This also applies to if you want to add more.

If you would like to add or rearrange elements in the scenario builder then see the scenarioBuilder.java file. Under the initComponents() method you can change the position as well as add new options. Please be careful of the panels however as certain components go into different panels.

To change what is being read and saved to the file then please navigate to the Editor.java class. Depending on what you need to change there are different methods that allow you to add, delete, save, convert the file to English, and convert the English back to file format. If you need to add new methods then make sure you add them in both of the convert sections and making sure that the information between them is interconnecting as you will see by the others already in there. If you need to change the way the List works then please do it through methods and not directly from the class somewhere. This is to help avoid any items getting shifted and or messed up within the list.

If you would like to edit the code in any way and you are not sure to what it does please feel free to contact us at anytime and we can assist you and also receive your input.