

Testplan

Jeder der folgenden Tests wurde zunächst in Google Chrome sowie Firefox durchgeführt. Über die Entwicklertools wurden auf beiden Browsern auch verschiedene Geräte getestet (GalaxyS5 und IphoneX).

Vorab: Es haben sich insbesondere im Bereich des User Interfaces grössere Probleme auf Mobile-Devices gezeigt, vor allem in Firefox. Würde das Projekt weiter getrieben, müsste man zunächst dieses Problem angehen.

Anmerkung: Einige der hier geschilderten Probleme mit dem UI konnten noch in letzter Minute gelöst werden (HTML-Validierung sei dank...)

1 User Interface

1.1 Testplan

ID	UI-Element	Erwartetes Aussehen
1	Statusmeldungen oben rechts im Browser ("App finished loading")	Gut lesbare Statusmeldung oben rechts, die nach einigen Sekunden ausgeblendet wird.
2	Menu links im Brwoser mit Auswahl der zu malenden Animationen und anderen Einstellungen	Vollständig sicht- und lesbares Menu. Menu kann ein- und ausgeklappt werden.

1.2 Durchführung

ID	Ergebnis	Bemerkungen
1	Google Chrome: OK auf allen Geräten Firefox, Galaxy und iPhone: Statusmeldung wird vergrössert und ist im Hochformat nicht mehr vollständig zu lesen.	Hier müsste wohl die Statusmeldung für Handys verkleinert werden.
2	Google Chrome, Galaxy und iPhoneX: "Close Controls"-Button wird nicht unten links am Menu fixiert, sondern schwebt frei und ist schwer zu erkennen. Gleiches Ergebnis wie auf Chrome	Möglicherweise verursacht durch das Überschreiben von Teilen des dat.gui-CSS.

1.3 Fazit

Im Bereich des User-Interfaces gibt es ganz klaren Nachholbedarf für Mobile-Devices. Sind die Probleme auf Chrome noch überschaubar, sind sie auf Firefox klar nicht akzeptabel. Wahrscheinlich müsste ganz auf die Bibliothek notify.js verzichtet werden und eine andere Lösung implementiert werden, was im Rahmen dieses Projekts aus Zeitgründen nicht möglich war.

Auch notify.js macht auf Mobile Probleme: Auf Google Chrome ist der Text kaum lesbar, auf Firefox ist er derart gross, dass er im Hochformat abgeschnitten wird.

2 Audioupload

2.1 Testplan

ID	Aktion	Erwartetes Ergebnis
1	Drag and Drop einer MP3-Datei	Erfolgreicher Upload mit entsprechender Statusanzeige oben links im Browser. Audiodatei wird mit Autoplay abgespielt und es erscheinen die HTML-5 Audiocontrols.
2	Drag and Drop eines PDF-Dokuments	Kein Upload und Anzeige einer Fehlermeldung oben rechts im Browser.

2.2 Durchführung

ID	Ergebnis	Bemerkungen
1	Google Chrome: ok Firefox: kein Autoplay. Audio muss manuell über die Controls gestartet werden.	Beim zweiten Versuch ging das Autoplay: Handelt es sich um eine Browsereinstellung? (Autoplay kann auch nerven)
2	Chrome und Firefox: OK	

2.3 Fazit

Keine grösseren Probleme in diesem Bereich.

3 Audiovisualisierung

2.1 Testplan

ID	Aktion	Erwartetes Ergebnis
1	Audio wurde mit Standardauswahl "Waveform" gedroppt.	Musik spielt ab, Wellenform wird in der gewählten Farbe abgespielt.
2	Auswahl wird auf "Animated Circle" geändert, Wellenform bleibt aber weiterhin angewählt.	Wellenform wird weiterhin gemalt, zudem erscheint auch der animierte Kreis. Das Untermenü zur Einstellung der Kreisanimation wird geöffnet.
3	Audio wird über die Audiocontrols pausiert.	Animationen werden eingefroren.

4	Farbslider werden verändert.	Sämtliche Animationen nehmen die gewählte Farbe an.
---	------------------------------	---

2.2 Durchführung

ID	Ergebnis	Bemerkungen
1	Google Chrome und Firefox: ok	
2	Beide Browser und alle Geräte OK. Kleines Manko: Der animierte Kreis geht auf Mobile im Hochformat links und rechts über die Bildschirmgrenze hinaus.	Der Kreisradius kann vom Benutzer angepasst werden, so dass er auf den Bildschirm passt.
3	Google Chrome und Firefox: ok	
4	Google Chrome und Firefox: ok	

2.3 Fazit

Keine grösseren Probleme in diesem Bereich.

4 HTML und CSS-Validierung

HTML und CSS wurden mit Hilfe der Seite <https://validator.w3.org/nu/> validiert. Dabei hat sich in letzter Minute einen Fehler in den meta-Tags gezeigt, der die oben geschilderten Probleme im User Interface teilweise gelöst hat (<meta name viewort ... anstatt <meta name="viewport" ...).

Um das von Javascript erzeugte HTML zu validieren, wurde an ausgewählten Stellen über den Befehl

```
console.log($(".html").html())
```

das HTML in die Konsole ausgegeben. Die zwei so gefundenen Fehler stammen beide von der externen Bibliothek „notify.js“.

5 Browserkompatibilität

Die Unterstützung von webgl und der Audio-api wurde mit „caniuse.com“ geprüft. Webgl wird von den meisten gängigen Browsern unterstützt, ausser Internet Explorer und Safari. Die Audio-API wird von Internet-Explorer nicht unterstützt.

Tests mit <https://www.browserstack.com/> wurden nur für ausgewählte Szenarien durchgeführt (mir lief während dem Testen die 30tägige Lizenz ab). Leider konnten auch keine Grafikanimationen getestet werden: Beim Droppen einer Datei will browserstack diese separat öffnen. Ich gehe davon aus, dass dies kein Fehler meiner Website ist. Es konnte deshalb nur getestet werden, ob die App überhaupt lädt (und damit auch, ob sie einen Audio- und einen WebGL-Kontext initialisiert) und ob das UI und die Notifikationen funktionieren.

Konfiguration	Ergebnis	Bemerkungen
Microsoft Edge auf Windows 10	Keine Fehler	
Firefox auf Windows 10	Keine Fehler	
Firefox Samsung Galaxy S7	UI ok Notify: Textmeldungen im Portätmodus gehen über den Bildschirmrand.	Gleiches Problem wie in den Entwicklertools schon gesehen.
Chrome Samsung Galaxy S7	ok	
Chrome iPhone XS	ok	

Testdurchführung:

Sonntag, 20. Oktober 2019

Unterschrift Testperson:

