

NetPaint – Software Architektur

WEB-TECHNOLOGIEN

VON FABIAN PUSZIES
MATR.-NR.: 530214

Inhaltsverzeichnis

1. Einführung
2. Interface
3. Verzeichnisstruktur
4. Objekte
5. Custom Element
6. Umgesetzte Stifte
7. Websocket Kommunikation
8. Fehlende Funktionen

1. Einführung

NetPaint ist eine Webanwendung die das Malen von Bildern erlaubt. Die Bilder können gemeinsam mit anderen Clients innerhalb von Räumen bearbeitet werden.

Die Applikation ist in Javascript umgesetzt und wurde mit Chrome v. 43 getestet.

2. Interface

Beim Layout habe ich besonderen Wert auf einfache Bedienbarkeit gelegt. Im Vordergrund der Applikation steht die weißte Leinwand auf welcher gezeichnet wird.

Am Rand befinden sich 2 Menüs, welche über einen einfachen Klick auf den Pfeil geöffnet werden können:

- Einstellungen
Hier werden allgemeine Einstellungen für das Canvas getroffen:
 - Neues Canvas erstellen
 - Bild speichern / laden
 - Einstellungen speichern / laden
 - Gruppen erstellen / beitreten
- Stifte
Hier lässt sich der Stift konfigurieren

Die Canvas wird immer zentriert dargestellt.

3. Verzeichnisstruktur

css/

Beinhaltet die css Dateien der Anwendung.

js/

Beinhaltet die init.js um den Drawer zu erstellen.

drawer/

Hier befinden sich alle für den Drawer nötigen Dateien

fonts/

In diesem Ordner befindet sich eine Icon Font. Die Icon Font wird z.B. für die Pfeile an den Menüs verwendet. Bezogen von: <http://getbootstrap.com/customize/>

4. Objekte

Drawer

Zur Initialisierung meines Plugins wird eine Instanz dieses Objekts erstellt.
Das Objekt dient der Verknüpfung von allen anderen Objekten.
Beim Erstellen dieses Objekts werden die Objekte der anderen „Klassen“ erstellt und Events zum Malen und Kommunizieren angelegt. Hier wird zum Beispiel der Submit Listener eines Form erstellt und entsprechend an andere Objekte kommuniziert.

- **Canvas**

In dieser „Klasse“ werden die Canvas Elemente erstellt und die eigentliche Zeichenlogik angespiegelt.

Diese Klasse speichert alle Klickpositionen, die sie übergeben kriegt, ab und führt Maloperationen auf der Leinwand durch. Außerdem werden hier in Abhängigkeit der aktuellen Klicks neue Canvas Elemente hinzugefügt oder wieder entfernt, um Ressourcen zu sparen.

Die eigentlichen Maloperationen werden in einer Unterklasse abgespeichert und nur noch aufgerufen. Dazu wird die Klasse Pen verwendet

- **Pen**

Das Pen Objekt dient dazu die eigentlichen Mal Operationen abzubilden und diese für andere User oder für die Zurück Operation abzuspeichern. Außerdem überwacht es das Stift Menu auf Änderungen.

Bei jeder Interaktion mit dem Canvas, wo eine Zeichenoperation hinzugefügt wird, wird bei Pen nach dem aktuellen Stift gefragt. Der aktuelle Stift erhöht sich nur, falls es Änderungen gibt und eine Interaktion durchgeführt wurde.

- **Communication**

Ein Objekt, welches zur Kommunikation von Clients dient. Das Objekt stellt eine Socketverbindung zwischen dem Client und dem Server her. Außerdem prüft es bei erhalten Nachrichten, ob es an den Client gerichtet sind.

Die eigentliche Programmierung der Kommunikation befindet sich in der Klasse Drawer, wo alle Operationen definiert sind.

Wenn Nachrichten an den Client gerichtet sind, werden sie über ein Dom Event auf dem Custom Element an die Drawer Klasse kommuniziert.

- **Storage**

Dieses Objekt dient für die Verwendung des Localstorages. Es können Bilder und Einstellungen gespeichert werden. Außerdem können verfügbare Einstellungen und Bilder abgefragt werden.

- **Interaction**

Namesbereich für Interaktionsfunktionen. Diese erstellen ein Callbackfunktionen, welche mit lokalen Variablen arbeiten können. Die einzelnen Funktionen werden über call und dem Element aufgerufen. Außerdem vermeidet es, dass eine Form einen Post Request durchführt.

- **HelpFunctions**

Dies ist lediglich ein Namesbereich für einige Hilfsmethoden. Z.B.:

- getRandomInt(min,max)
- readForm()

5. Custom Element

Das Custom Element erstellt ein Div, welches in der Mitte des Bildschirms dargestellt wird, wenn man über die Methode setContent Text hinzufügt. Per Standard ist das Element nicht sichtbar.

Das Element kann durch einen Click wieder geschlossen werden. Wenn dies nicht passiert, wird es nach 3 Sekunden automatisch entfernt.

6. Umgesetzte Stifte

Grundlegende Funktion aller Stifte über das Stifte Einstellungsmenu

- Einstellung der Stiftbreite
- Einstellung der Farbe
- Einstellung der Art der Linienverbindung

Standard Maler

Ein ganz gewöhnlicher Stift. Bei einer Drag Bewegung verbindet er den aktuellen Punkt mit dem letzten.

Random Maler

Der Random Maler. Ist ein Zufallsmaler. Er bemalt die Klickfläche lediglich zu 50% ähnlich einem Spray, jedoch tut er dies mit zufälligen Farben als Viereck.

Spray

Ist ein Spray, welches lediglich zufällige Teile der Klickfläche in gewählter Farbe bemalt.

Radierer

Funktioniert, wie der gewöhnliche Stift, allerdings ist hier die Farbe Weiß fest gewählt.

7. Websocket Kommunikation

Die Websocket Kommunikation dieser Software findet mit JSON Strings, welche bei Empfang geparkt werden, statt.

Die Pakete dürfen die Größe von 200 Zeichen nicht übersteigen.

Die Pakete sind wie folgt aufgebaut:

kennung	message	operation	from	group
---------	---------	-----------	------	-------

Alle Values dieser Nachricht sind verschlüsselt. Der „from“ Wert ist die UserId, welche nach dem Verbinden, vom Server übertragen wird. Group ist nicht gesetzt, wenn der User in keiner Gruppe ist.

Mit Gruppe verbinden und Einstellungen übertragen

Wenn ein Client die Webanwendung öffnet. Löst er einen Broadcast an alle anderen Clients aus. In diesem Broadcast fragt er nach Gruppen.

Sollte ein anderer Client in einer Gruppe sein, antwortet er mit „setGroup“ und dem Gruppennamen. „setGroup“ wird außerdem ausgelöst, wenn ein Client eine neue Gruppe erstellt.

Das fügt dann eine neue Gruppe im Select Feld der Gruppen hinzu, falls diese noch nicht vorhanden ist. Wählt man als Client nun diese Gruppe im Select Feld aus, tritt man ihr bei und sendet allen Gruppenmitgliedern „joinGroup“ mit seiner UserId.

Die „joinGroup“ Operation wird nur ausgeführt, wenn das Feld „group“ der eigenen entspricht. Die anderen Clients antworten mit den aktuellen Einstellungen der Gruppe mit dem Operationsname „setGroupOptions“ außerdem wird im Feld Message die Empfänger ID eingetragen.

Sobald der beitretende Client nun „setGroupOptions“ erhält, übernimmt er die Einstellungen

aus dem Feld Message und erstellt ein neues Canvas. Außerdem wird der User, welcher zuerst geantwortet hat, als Kommunikationspartner gewählt. Von diesem werden noch alle bereits gemachten Maloperationen abgefragt.

Kommunikation innerhalb von Gruppen

Sobald eine Touch- oder Klickbewegung bei einem Client durchgeführt wurde. Sendet er einen Broadcast an alle anderen Clients mit der Maloperation inkl. Des Stifts.

Sollte ein Empfänger dieser Nachricht nun innerhalb der gleichen Gruppe sein, übernimmt er die Klickposition und den dazugehörigen Stift.

Eine Liste der verfügbaren Operationen:

Operationsname	Funktion
getGroups	Frage nach verfügbaren Gruppe, alle User, die in einer Gruppe sind, antworten.
setGroup	Antwort auf verfügbare Gruppen
addClick	Einfügen von Zeichenoperation, falls in der gleichen Gruppe
joinGroup	Ein Benutzer tritt der Gruppe bei
setGroupOptions	Optionen werden beim Gruppenbeitritt zugesendet
getClicksDone	Bereits gemachte Klicks werden beim Gruppenbeitritt angefragt
setClicksDone	Bereits gemachte Klicks erhalten
setUserNumber	Die User Number wurde vom Server erhalten
revertStep	Letzte Touch- oder Klickbewegung entfernen

8. Fehlende Funktion

- Gruppen sind nicht in der Anzahl der Teilnehmer begrenzt
- Hintergrundfarbe ist nicht einstellbar / Nur Hintergrundbild ist einstellbar
- Individuelle Stifte werden nicht persistent gespeichert
- Kein Passiv Modus (Design grundsätzlich darauf ausgelegt)
- Keine Unterbrochen Funktion