

Konzipieren und entwickeln Sie eine Sprachlern-App unter Berücksichtigung folgender Anforderungen:

- Zielgruppe sind **deutschsprachige Nutzer**, die **Spanisch** in einem Anfängerkurs lernen wollen.
- Beim Start des Kurses in der Sprachlern-App wird zunächst ein Satz auf Deutsch angezeigt (bspw.: "Ich heiße Gabriel").
- Ebenso werden die einzelnen Wörter der Übersetzung auf der zu erlernenden Sprache angezeigt, aber in einer zufälligen Reihenfolge (die Übersetzungswörter für oben genannten Satz in zufälliger Reihenfolge wären bspw.: "Gabriel" "llamo" "Me").
- Der Nutzer muss nun die übersetzten Wörter in der richtigen Reihenfolge anklicken. Der erste Klick muss auf das Wort an erste Stelle des übersetzten Satzes gesetzt werden, der zweite Klick auf das Wort an zweiter Stelle usw...
- Klickt der Nutzer ein Wort richtig an, dann gibt es einen Punkt, das Wort wird in einem Übersetzungsbereich angezeigt (bspw.: "Me"). Der Nutzer kann danach ein anderes Wort anklicken.
- Klickt der Nutzer wieder ein Wort richtig an, dann gibt es einen weiteren Punkt, das Wort wird ergänzend im Übersetzungsbereich angezeigt ("Me llamo"). Der Nutzer kann danach ein anderes Wort anklicken.
- Klickt der Nutzer ein falsches Wort an, gibt es einen Punkt Abzug, im Übersetzungsbereich werden nur die bisher richtigen Wörter angezeigt.
- Das Punktekonto des Nutzers darf nicht in den Minusbereich gehen. Der aktuelle Punktestand ist für den Nutzer immer sichtbar.
- Sobald die Übersetzung vollständig ist wechselt die Anwendung zum nächsten Übungssatz.
- Wenn alle Übungssätze abgeschlossen wurden wird dem Nutzer der finale Punktestand angezeigt und zum Abschließen des Kurses gratuliert.
- Beim Start bzw. Neustart der Anwendung werden die Übungssätze in einer zufälligen Reihenfolge durchgeführt. Ebenfalls zufällig werden immer die Worte in Ihrer Reihenfolge dargestellt.
- Sie können selbst die Übungssätze definieren. Ebenso können Sie selbst entscheiden, wie Sie Worte und Satzzeichen trennen (bspw. bei "¡La vida local!" können die Ausrufezeichen separat oder als Teil der Worte verstanden werden).
- Die Anwendung benötigt keine speziellen Animationen zwischen den Darstellungen. Die Inhalte können einfach "sofort-artig" erscheinen.
- Die Sprachlern-App kann in drei **Schwierigkeitsstufen** gespielt werden:
 - 0. Leicht: der Kurs besteht aus 5 Übungssätzen.
 - 1. Mittel: der Kurs besteht aus 10 Übungssätzen.
 - 2. Schwer: der Kurs besteht aus 15 Übungssätzen.
- Die Anwendung ist **optimiert für eine Smartphone Darstellung** im **Portrait-Format** (ca. 375 x 667 Pixel).

Beurteilung

Note der End-Abgabe basiert auf den dafür ausgefüllten Anforderungs-Punkten (siehe unten). Bis zu ein Punkt darf dabei nicht erfüllt sein um die 1 zu erhalten.

Jeder weitere fehlende Punkt verringert die Note um 1. (4.0 >= Bestanden)

Jedes erfüllte optionale Ziel erhöht die Note um 1. (Maximale Bestnote: 1).

Der Kurs wird im Studi-Portal mit "bestanden" / "nicht bestanden" bewertet.

Tipps

Machen Sie sich ein Konzept!

Versuchen Sie den Anlauf der Sprachlern-App in einem Flowchart/Anwendungsfalldiagramm darzustellen. An welchen Punkten werden Entscheidungen gefällt? Muss der Computer gewisse Sachen abfragen? Wann muss der Nutzer interagieren, wie interagiert er, was sind seine Optionen? Welchen Ablauf hat der Computer, wie wählt er einen Satz aus? Wie werden die Inhalte dargestellt? Welche Elemente sind interaktiv und können vom Nutzer ausgewählt werden? Machen Sie sich darüber Gedanken, wann sich die Darstellung für den Nutzer ändert.

Anforderungspunkte

Gesamt: 21. (21 = 1.0, 20 = 1.0, 19 = 2.0, 18 = 3.0, 17 = 4.0, <= 16 = NB)

Nach folgenden Kriterien wird Ihre Abgabe bewertet:

Konzeptentwurf

- In einem Flussdiagramm haben Sie den grundsätzlichen Interaktionspfad der Anwendung dargestellt (Interaktion des Nutzers und Verhalten des Systems). Den Papier-Entwurf bitte als Scan / Foto digitalisieren und beilegen.
- Bilden Sie ebenso in einer Skizze die Darstellung des GUIs ab und legen Sie die Darstellung Ihrer Abgabe bei. Sie können ein

Wireframing-Tool nutzen oder mit Papier und Bleistift arbeiten (Papier-Entwurf bitte wieder scannen).

HTML

- Ein schlichter Footer mit Ihrem Namen und Matrikelnummer.
- Semantische HTML-Tags wurden genutzt, Tags werden sinnvoll eingesetzt.

CSS

- Die Fläche, auf der die Übungssätze dargestellt werden, besitzt eine passende Hintergrundfarbe (oder Hintergrundtextur).
- Die Inhalte (Übungssatz, Übersetzungswörter, Übersetzungsbereich usw.) sind sinnvoll auf dem Bildschirm verteilt, voneinander unterscheidbar und überlagern sich nicht.
- Die einzelnen Übersetzungswörter sind als auswählbare Elemente erkennbar.
- Die Darstellung ist für oben genanntes Endgeräte-Format optimiert.

Programmierung

- Das Skript erzeugt keine Fehler in der Konsole.
- Ein Übungssatz mit seiner Übersetzung ist auf programmatischer Ebene sinnvoll als Objekt strukturiert.
- Das Skript bildet, je nach ausgewählter Schwierigkeitsstufe, eine bestimmte Anzahl von Übungssätzen, die nacheinander ausgegeben werden.
- Die Reihenfolge der Übungssätze wird zufällig generiert und variiert daher bei jedem Start. Jeder Übungssatz kommt nur einmal pro Kursdurchlauf vor.
- Die Reihenfolge der Übersetzungswörter wird ebenfalls zufällig generiert.
- Das Skript erzeugt für jeden Übungssatz und den Übersetzungswörtern eine Darstellung mit HTML-Elementen.
- Die einzelnen Übersetzungswörter können vom Nutzer ausgewählt werden. Ein mehrfaches Auswählen eines Wortes ist möglich.
- Das Skript überprüft bei Auswahl eines Übersetzungswortes, ob dieses Wort in der richtigen Reihenfolge ausgewählt wurde. Dafür muss das Skript für ein ausgewähltes Wort die entsprechende Stelle im übersetzten Satz kennen.
- Das Skript übernimmt das Übersetzungswort in den Übersetzungsbereich, wenn das Wort an der richtigen Reihenfolge geklickt wurde und zählt den Punktestand des Nutzers um einen Punkt hoch.
- Das Skript reduziert den Punktestand des Nutzers um einen Punkt, wenn das ausgewählte Wort nicht korrekt ausgewählt wurde und gibt eine einfache Fehlermeldung (per Alert) aus.
- Das Skript zeigt den nächsten Übungssatz an, sobald für den aktuellen Übungssatz die Übersetzung vollständig ist.
- Das Skript zeigt am Ende, wenn alle Übungssätze abgeschlossen wurden, eine Zusammenfassung mit dem finalen Punktestand an.

Annotationen

- Erklären Sie Ihren Code. Die Skript-Datei sollte mit Kommentarblöcken ergänzt sein, um relevante Anwendungsblöcke und Anweisungen beschreiben zu können.

Optional Diese Punkte müssen NICHT erfüllt werden! Sie fließen jedoch positiv in die Bewertung mit ein (siehe oben).

- Visuell ansprechende Darstellung der Übungssätze und Übersetzungswörter.
- Das Auswählen von Wörtern und das positive Feedback an den Nutzer (richtiges / falsches Wort), sowie das Wechseln von einem Übungssatz zum nächsten ist animiert.
- Relevante Aktionen in der Anwendung werden mit Sounds unterstützt (die Auswahl von einem Wort; das Feedback zur Auswahl falsch / richtig; nächster Übungssatz; Ende des Kurses).
- Um es für den Nutzer etwas schwerer zu machen werden zu den Worten, die im tatsächlichen Übersetzungssatz auch vorkommen, zusätzlich auch andere Worte angezeigt, die nicht vorkommen.
- Zum Anwendungsstart kann der Nutzer, neben Spanisch, auch eine andere Sprache wählen, die geübt werden soll. Besonders schön wäre eine Erweiterung, um Ukrainisch zu lernen.
- Neben der Punktezahl wird ebenfalls der aktuelle Fortschritt als Text-Element (bspw.: "Übungssatz 3 von 15") und in Form eines Balkens (Progressbar) angezeigt.

Abgabe

Die Abgabe der Arbeit erfolgt diesmal nicht über Github, sondern hier im Felix-Kurs, auf der Unterseite "Abgabe Endaufgabe". Verpacken Sie Ihr Ergebnis als ZIP-Datei und laden Sie Ihre Abgabe in Ihren Teilnehmerordner in Felix hoch (*). Der Abgabebereich ist bis zur Abgabefrist zugänglich. Ihre ZIP-Datei sollte die Größe von 300MB nicht überschreiten.