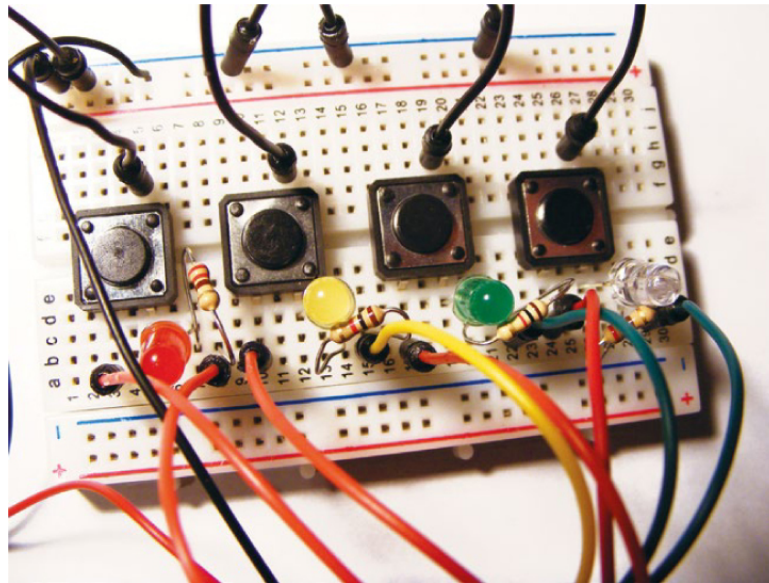


## Projektaufgabe — Gedächtnis-Trainer



Wir wollen die Programmierung und die Techniken, die wir im Projekt kennenlernen, an einem praktischen Beispiel ausprobieren. Dabei liegt unser Fokus auf der Software und ihren Strukturen und insbesondere auf einer **nachvollziehbare Dokumentation**.

### Aufgabe — Gedächtnis-Trainer

Es soll ein *Gedächtnis-Trainer* entstehen, der auf Knopfdruck mittels LEDs ein Spiel startet, das unser Erinnerungsvermögen trainiert. Auch das Ergebnis soll über LEDs visualisiert werden. Der Gedächtnis-Trainer besitzt drei Knöpfe und ihnen zugeordnet drei möglicherweise verschiedenfarbige LEDs.

Der Trainer soll wie folgt benutzt werden: Nach dem Einschalten zeigt der Gedächtnis-Trainer eine Einschalt-Animation und das Spiel beginnt. Es geht über mehrere Runden, begonnen in Runde 1, in der nur ein LED-Aufleuchten gezeigt wird. In jeder Runde lässt der Trainer eine Sequenz von LEDs aufblinken. Aufgabe des Spielers ist, diese Sequenz anhand der zugehörigen Tasten nachzudrücken. Ist der Spieler nicht erfolgreich so ist das Spiel beendet, eine Abschluss-Animation wird gezeigt und ein neues Spiel beginnt. Ist der Spieler erfolgreich so wird eine Bestätigungs-Animation gezeigt und die nächste Runde beginnt: der bestehenden Sequenz wird ein weiteres, zufälliges LED-Aufleuchten hinzugefügt, die bisherige Sequenz bleibt unverändert. Dadurch wird die Sequenz in jeder Runde länger und das Merken der Sequenz für den Spieler immer schwerer. *Wie lang kann die Sequenz werden, bis Du sie Dir nicht mehr merken kannst?*

Beispiel: Der Spieler startet das Spiel durch Einschalten. Einschalt-Animation

- Runde 1: Der Trainer zeigt *rot*. Der Spieler drückt richtig den Taster der roten LED. Bestätigungs-Animation
- Runde 2: Der Trainer zeigt *rot, grün*. Der Spieler drückt richtig den Taster der roten, dann der grünen LED. Bestätigungs-Animation
- Runde 3: Der Trainer zeigt *rot, grün, grün*. Der Spieler drückt richtig den Taster der roten LED, der grünen LED, dann fälschlich den der roten LED. Abschluss-Animation. Das Spiel ist beendet und ein neues Spiel beginnt.

## Vorgehen

Die Arbeitsweise soll *bottom-up*-orientiert und *iterativ* sein: Zunächst soll ein minimales Programm entstehen, das lediglich die grundlegende Funktionalität realisiert und so unter Beweis stellt, dass die Hardware angemessen angesteuert werden kann. Animationen etwa können hier leer sein. Diese Version soll dann unter Beibehaltung der Zwischenversionen in einem Repository schrittweise ausgebaut werden. Am Ende steht ein Prototyp des elektronischen Gedächtnis-Trainers, der die oben beschriebene Funktionalität und möglicherweise darüber hinaus weitere Features hat. Eine begleitende **Entwickler-Dokumentation** beschreibt den grundsätzlichen Aufbau des Programms (etwa in Form eines Flussdiagramms). Auch sie wird laufend fortgeschrieben.

## Hardware

Die Hardware des elektronischen Gedächtnis-Trainers soll aus etwa einem Arduino Uno, den Tastern und Leuchtdioden (mit Vorwiderständen) bestehen. Wie eine mögliche entsprechende Hardware als auch das Software-Grundgerüst aussehen kann, besprechen wir in den Projekt-Treffen bzw. in der AG-Smart-Technology.

Unser Augenmerk liegt dabei nicht auf der Hardware und so sollte nicht viel Engagement in den mechanischen oder elektrischen Aufbau gesteckt werden. Ein Prototyp auf einem Steckbrett ist vollkommen ausreichend. Die **Hardware-Dokumentation** sollte allerdings natürlich den Aufbau und die von der Software erwartete Verdrahtung beschreiben (inklusive Schaltplan).

Ein Gehäuse benötigen wir nicht. Die Stromversorgung darf über den Arduino erfolgen (per USB oder über ein Netzteil).

## Software

Für die Software soll neben dem Programm auch essentielle **Entwickler-Dokumentation für Analyse, Entwurf, Implementierung, Test und Ablauf** geben. Alle Artefakte (alles was schöpferisch für das Projekt entsteht) soll in einem Repository versionsverwaltet werden. Um die Aufgaben zu koordinieren könnt Ihr natürlich zusätzlich auch gerne Trello verwenden.

Das Projekt soll **einzel**n bearbeitet werden. Jeder entwickelt eine **eigene Version der Software und der zugehörigen Dokumentation** in einem eigenen Repository. Hardware darf geteilt werden, so dass (nach Neukonfiguration) das eine oder andere Programm auf der Hardware abläuft. Um auch unabhängig von der Hardware entwickeln zu können, kann die Tasten-Eingabe und die LED-Ausgabe ggf. auch simuliert werden.

Folgendes Vorgehen wird vorgeschlagen:

1. GitHub-Account (sofern noch nicht vorhanden) erstellen und ein Repository für das Projekt anlegen. Dafür muss man sich bereits einen griffigen Namen für das Projekt ausdenken (Brainy3, RememberMe, ...). Dabei auch ein informatives **README.md** schreiben und eine passende Verzeichnisstruktur (Quellcode, Dokumentation, ...) definieren.
2. Programmstruktur planen. Wie sieht die genaue Benutzung aus (Use-Case und Use-Case-Diagramme)? In welchen Zuständen kann sich der elektronische Gedächtnis-Trainer befinden (Zustandsdiagramm). Gibt es mehrere Komponenten, die zusammenwirken (Komponentendiagramm)? Was sind Ihre Aufgaben? Wie wird der Zufall erzeugt? Generiert der Gedächtnis-Trainer nach dem Einschalten immer wieder die gleichen Sequenzen? Müssen die Taster entprellt werden? ... Ergebnis dieser Überlegungen ist eine **anfängliche Entwickler-Dokumentation** mit Skizzen, in dem die Anforderungen, die Funktionsweise, Ablauf und Test des Gedächtnis-trainers dargestellt sind. Diese anfängliche Entwickler-Dokumentation wollen wir gemeinsam ansehen und diskutieren.

3. Minimale Programm-Version (Prototyp) realisieren, um die Hardware anzusprechen und die Grundfunktionalität bereitzustellen. **Benutzer-Dokumentation** für diese minimale Version erstellen.
4. Weitere Funktionalität, etwa ein weiteres LED/Taster-Paar realisieren. **Hardware-, Benutzer- und Entwickler-Dokumentation** für diese erweiterte Version aktualisieren.
5. Optional weitere Features (etwa weitere Animationen, eine Audio-Ausgabe, ...) planen, realisieren und dokumentieren.
6. Präsentation der Ergebnisse

Bitte trennt, nachdem Ihr Euren Prototyp gebaut habt (3), das Ansprechen der Hardware (Taster, LED) und die Gedächtnis-Trainer-Logik voneinander. Das Ansprechen der Hardware soll in eine projekt-spezifische Bibliothek ausgelagert werden. Definiert dazu bitte geeignete Schnittstellenfunktion und verwendet sie im Hauptprogramm bzw. im Logik-Teil.

## Abschluss

Die Projekt-Ergebnisse sollen am Montag, dem 10. Januar 2022 vorgeführt werden. Dazu pitcht jeder sein Projekt in einem Kurzvortrag (3 Folien), demonstriert seinen elektronischen Gedächtnistrainer und steht Rede und Antwort. Der dann jeweils letzte Commit in Euren Repositories ist der Abgabestand.