

Projektarbeit Web-Technologien WiSe 24/25 (Stand: 10.1.2025)

Projektarbeit:

- **Zulassung:** mind. fünf Übungen bestanden und mind. fünf Code-Reviews durchgeführt (Details siehe Moodle).
- Bearbeitung in **Einzelarbeit, Zweier- oder Dreiergruppen** möglich.
- Je nach Programmier-Kenntnissen und Interesse kann der Fokus auf das **Frontend und die clientseitige Entwicklung mit JavaScript** (oder TypeScript) gelegt werden **oder auf die serverseitige Entwicklung mit PHP** (Details siehe nächste Folien).
- **Gruppen** müssen sowohl die clientseitige wie die serverseitige Variante umsetzen.
- Die Projektarbeit ist eine **Übung zur Webprogrammierung** („Vanilla“-JS bzw. PHP), deshalb ist es nicht sinnvoll, komplexe Frameworks (wie Symfony oder React) zu verwenden (bitte Fragen Sie mich im Vorfeld, wenn Sie das trotzdem beabsichtigen). Sinnvolle Hilfsmittel: GitHub Copilot, ChatGPT, einfache Layout-Frameworks (Bootstrap, ggf. ChartJS).

In der **Sitzung im WiSe am 14.1.2025** wurden diese Folien ausführlich besprochen. Wenn Sie an diesem Termin nicht anwesend sein konnten oder weitere Fragen haben: ich helfe gerne, schreiben Sie mir eine Mail und/oder vereinbaren Sie einen Sprechstundentermin.

Aufgabenstellung

Erstellen Sie für den Kunde: „Privatbank Mustermann & Co“ eine Webanwendungen/Webseiten bestehend aus folgende drei Teilen:

1. Homepage / Startseite

ohne „Inhalt“, nur mit „Lorem Ipsum“ (siehe Beispiel folgende Folie), wichtig:
Responsiven Design (Tipp: Bootstrap)
Dateiname: index.php oder index.html

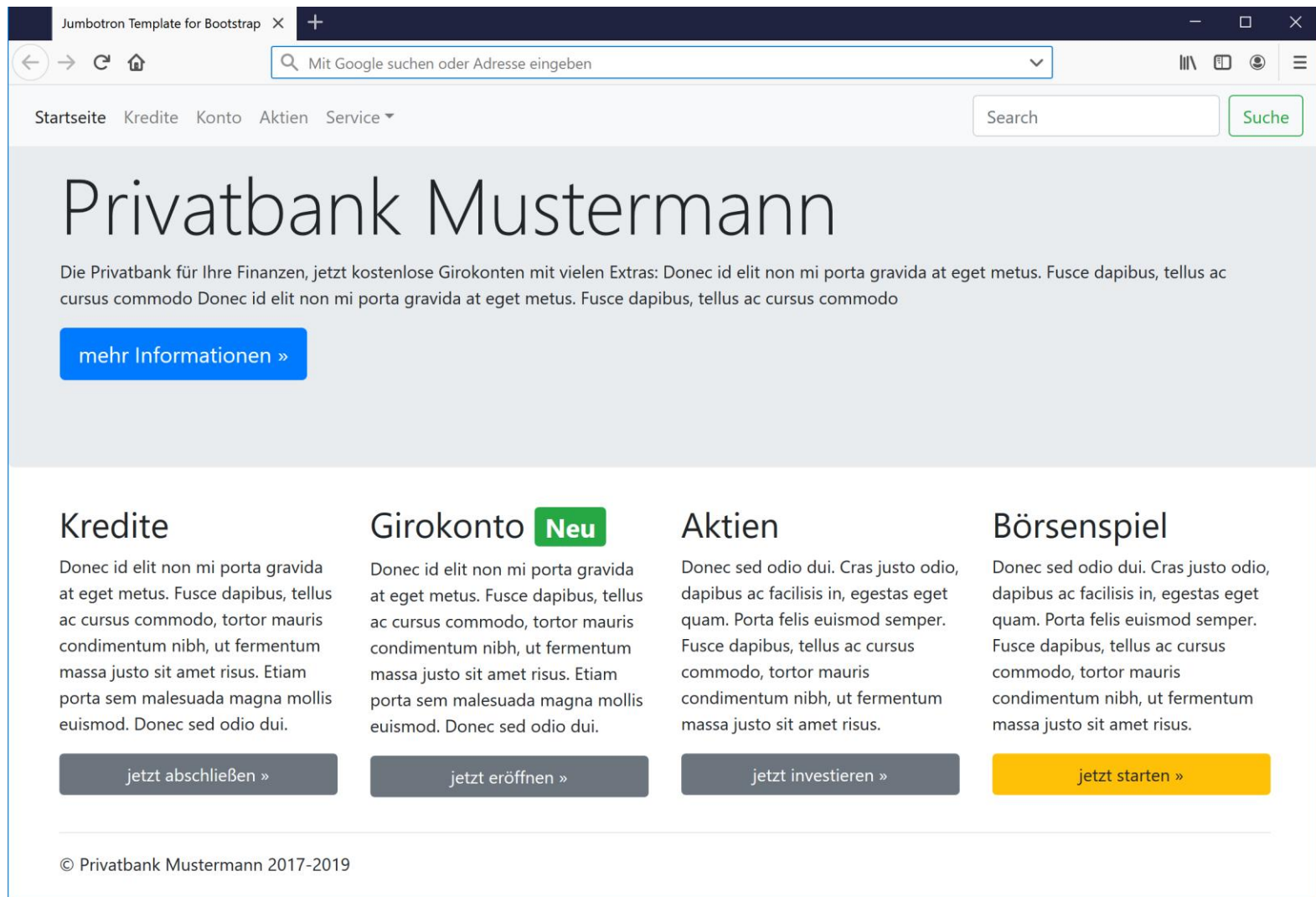
2. eine Seite „Registrierung Börsenspiel“

die von der Startseite verlinkt wird
Dateiname: registrierung.html (oder registrieung.php)
(ggf. JavaScript-Datei: registrierung.js)

3. eine Webanwendung „Börsenspiel“

die von der Seite „Registrierung“ verlinkt wird
Dateiname: boersenspiel.html (oder boersenspiel.php)
(ggf. JavaScript-Datei: boerse.js*)

Vorschlag Startseite (Teil 1)



NICHT (!!!) ERFORDERLICH

- aufwendige Layout, Logos oder Grafiken (bei Grafiken: Copyright beachten; Quellenangaben). Sie übernehmen die Aufgabe von Webprogrammiererinnen und -programmierer, nicht von Grafikerinnen/ Grafiker und/oder Designerinnen / Designern!
- komplette Umsetzung aller denkbaren Funktionen einer Bank-Website – erstellen Sie nur die drei Seiten, die beschrieben sind!
- aufwendiger (Text-)Content und Berücksichtigung sonstiger Aspekte wie Impressum, AGB, rechtliche Vorgaben etc. – Schreiben Sie nur „Lorem Ipsum..“

Teil 2: Registrierungs-Seite

- a) Die Userin/der User muss sich mit folgenden Daten registrieren:
- Anrede, Vor- und Nachname (Pflichtfelder)
 - Zwei Felder „E-Mail“ und „bitte E-Mail wiederholen“ (Pflichtfeld; Fehlermeldung, wenn die Eingabe beider Felder nicht identisch ist)
 - Geburtsdatum (Fehlermeldung, wenn User nicht mind. 18 Jahre)
 - PLZ des Wohnorts (Fehlermeldung, wenn PLZ keine fünfstellig Ziffernfolge)
- b) Es sollen möglichst benutzerfreundliche Fehlermeldungen ausgegeben werden.
- c) Bei erfolgreicher Registrierung wird eine entsprechende Meldung angezeigt, das Registrierungs-Formular ausgeblendet und eine Taste „weiter“ angezeigt, die auf das Börsenspiel (nächste Aufgabe) verlinkt.
- d) Die Daten werden bei entweder serverseitig validiert und in einer Datenbank gespeichert (wenn Schwerpunkt „PHP“ bzw. für Gruppenarbeiten) oder clientseitig validiert und in einem „local storage“ gespeichert (wenn Einzelarbeit mit Schwerpunkt „JavaScript“).

Teil 3: Aufgabenstellung „Börsenspiel“

Börsenspiel

Die Bank möchte ein kleines „Serious-Game“ (Lernspiel) Rund um das Thema "Börse" entwickeln lassen, in dem die Grundlagen (und Risiken) des Aktienhandels spielerisch erfahren werden können.

Wichtig: Im Fokus dieser Arbeit steht die Entwicklung eine Webanwendung, ausführliches Domänenwissen (Börse, Aktienhandeln etc.) ist nicht erforderlich und ist nicht Teil der Projektarbeit (und selbstverständlich auch nicht Teil der Benotung). Im Rahmen des Prototyps sind nur die Umsetzung einfachster Grundlagen (Geld- und Briefkurs, Gebühren) notwendig, Details siehe folgende Folie.

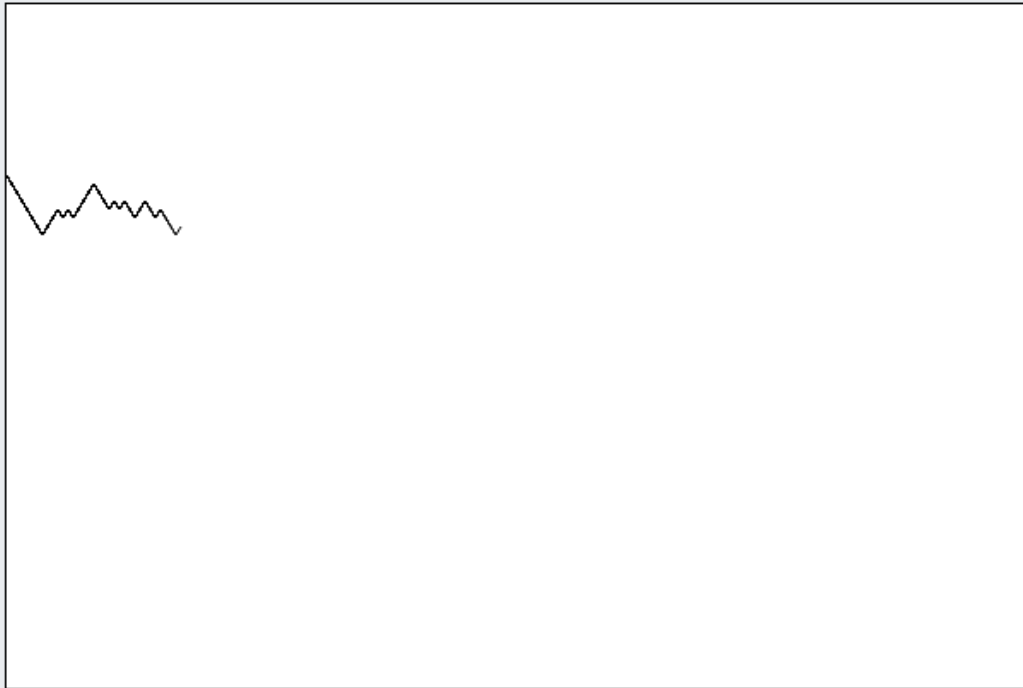
Neben den **Vorgaben** auf den folgenden Folien muss bei Einzelarbeiten und Zweiergruppen mindestens eine **eigene Spielidee** (z. B. „Börsencrash“, detaillierte Auswertung, Kreditaufnahme) eingebaut werden, bei Dreiergruppen mindestens drei eigene Ideen.

Teil 3: clientseitige Programmierung (JavaScript)

- a) Spielbeginn: Visualisierung einer Börsen-Chart (siehe Beispiel-Screenshot nächste Folie) für eine Aktie der "Mustermann AG", Startkurs z. B. 100 Euro (Briefkurs) und 99 Euro (Geldkurs), d.h. ein Spread von 1 Euro. Spielgeld: z. B. 50.000 Euro
- b) Im Sekundentakt kommt es zur zufälligen Kursänderung (Brief und Geldkurs) z. B. +/- 0,01 bis 0,05 Euro, dies wird auf dem Chart visualisiert (Tipp: Canvas). Wahrscheinlichkeit jeweils 50%
- c) Fällt oder steigt die Aktie dreimal hintereinander, ist ein „Bärenmarkt“ (=Phase fallender Kurse) bzw. ein „Bullenmarkt“ (=Phase steigender Kurse), dies wird entsprechend angezeigt; die Chancen, dass die Aktien weiterhin fällt oder steigt verdoppeln sich dann (d. h. 75% zu 25%).
- d) Durch Klick auf "Kaufen" wird eine zuvor auswählbare Anzahl an Aktien zum Briefkurs erworben. Als Orderprovision (Gebühr) wird Euro 4,95 zzgl 0,25% vom Orderwert berechnet, minimal Euro 9,99, maximal 59,99 Euro. Gebühren sowie Depotbestand (Anzahl gekaufter Aktien) wird angezeigt.
- e) Durch Klick auf "Verkaufen" wird eine zuvor auswählbare im Depot (Besitz) befindliche Anzahl an Aktien zum Geld-Kurs verkauft (Orderprovision wie bei Kauf).
- f) Angezeigt werden soll: Anzahl des aktuellen Spielgeld-Guthaben, aktueller Gewinn/Verlust, aktuelle Depotbestand (Anzahl Aktien)
- g) Spielende nach einer bestimmten Zeit (z. B. 10 Minuten) oder Klick auf die Taste "Spiel beenden"; Anzeige Gewinn/Verlust
- h) Eine eigene Spielidee (siehe Folie zuvor); bei Dreiergruppen: drei Spielideen!

Beispiel: einfache Chart

Börsenspiel



Spiel starten

Aktuelle Kurs: Brief 70,0 Euro - Geld 69,0 Euro

eine Aktien kaufen

eine Aktien verkaufen

Kontostand: 50.000 Euro Anzahl Aktien im Besitz: 0 Aktien Restzeit: 300 Sekunden

Entwurf der Chart – für die Ausarbeitung sind weitere Punkte notwendig (z. B. Anzeige Gewinn/Verlust, Gebühren, etc.)

Teil 3: serverseitigen Programmierung (PHP)

- a) Das Registrierungs-Formular enthält zusätzlich ein Feld zur Wahl des „Passwort“ und „Passwort wiederholen“.
- b) Login in Musterdepot mit Mail und Passwort, Spielgeld zu Beginn: 50.000 Euro
- c) Im Musterdepot sind zehn exemplarische Aktien zu handeln (wählen Sie zehn Unternehmen z. B. aus dem DAX oder Dow Jones oder erfinden Sie Namen wie „Mustermann AG“). Startkurs jeweils 100 Euro (Briefkurs) und 99 Euro (Geldkurs)
- d) Durch Klick auf "Kaufen" wird eine zuvor auswählbare Anzahl an Aktien zum Briefkurs erworben. Als Orderprovision (Gebühr) wird Euro 4,95 zzgl. 0,25% vom Orderwert berechnet, minimal Euro 9,99, maximal 59,99 Euro. Gebühren sowie Depotbestand (Anzahl gekaufter Aktien) wird angezeigt.
- e) Durch Klick auf "Verkaufen" wird eine zuvor auswählbare im Depot (Besitz) befindliche Anzahl an Aktien zum Geld-Kurs verkauft (Orderprovision wie bei Kauf).
- f) Ansichten:
 - Depotbestand (Anzahl des aktuellen Spielgeld-Guthaben, aktueller Gewinn/Verlust, aktuelle Depotbestand, Anzahl Aktien)
 - Übersicht Aktienkurse (der zehn handelbaren Aktien)
 - Detailansicht einer Aktie (Kurse der letzten zehn Tage).
 - Orderbuch (Ansicht aller Käufe und Verkäufe)
- g) Eine eigene Spielidee (siehe Folie zuvor); bei Dreiergruppen: drei Spielideen!

In Einzelarbeit: Alle zehn Aktienkurse ändert sich einmal täglich zufällig um +/- 0,01 bis 2,5% (dies kann auch manuell in der Datenbank erfolgen). Es ist nicht zwingend erforderlich, ein Chart zu erstellen. Stattdessen ist eine Anzeige "Vortageskurs" und "Tageskurs" ausreichend, in der Detailansicht der Aktie die Kurse der letzten 10 Tage.

In Teamarbeit: Chart erforderlich, wie unter „clientseitig“ beschrieben!

Allgemeine Hinweise (für client- und serverseitige Projekte)

- Neben den Anforderungen auf den Folien zuvor ist bei Einzelarbeiten und Zweiergruppen eine **eigene Spielidee**, bei Dreiergruppen sind drei Spielideen umzusetzen (siehe Beispiele auf Folie 6).
- Bei Gruppenarbeiten ist der Einsatz der **Versionsverwaltung Git** erforderlich (auf GitHub, GitLab oder am besten dem THM-GitLab).
- Hinweis: Die locker formulierten "Anforderungen" auf den Seiten zuvor sind mit Absicht nicht als User Stories, Use Cases o.ä. formuliert, so dass dies auch ohne Kenntnisse dieser Methoden verständlich ist. Bei Fragen zu den Anforderungen (und auch bei allen anderen Fragen) können Sie mich jederzeit und gerne kontaktieren (siehe letzte Folie).

Abgabe: 28.3.2025

- **Einzelarbeit: Quellcode** und sämtliche Dateien als ZIP-Paket; **bei Gruppen: Einladung in GitHub bzw. GitLab-Repo**
- **Video-Projektdokumentation:** Studierenden präsentieren persönlich das Ergebnis des Programmentwurfs als kompakte Video-Zusammenfassung, Dauer ca. 3-6 Minuten (pro Studierenden), Aufbau:
 - Begrüßung/Einführung: Folie mit Namen, Matrikelnummer, Datum etc. und (kleines) Video-Bild des Studierenden (z. B. mit PowerPoint, ca. 30 Sekunden)
 - Demo der Webanwendung: Bildschirmaufzeichnung mit Audio-Kommentar (z. B. mit OBS-Studio, obsproject.com; ca. 2-5 Minuten)
 - Blick in den Quellcode: kritische oder besonders gut gelöste Stellen (ca. eine Min.)
 - Kurzes Fazit / kritische Reflexion ca. 1-2 Minuten, weitere Ideen
 - Projektteam, Aufgabenverteilung, Projektplanung
 - Vorgehen (ablauforientiert: „Zuerst wurde ... dann ...“).
 - Begründung, warum Bibliotheken/Frameworks (nicht) verwendet wurden.
 - Beschreibung der „Toolchain“ (Entwicklungswerkzeuge, Hilfsmittel, Tools)
 - Beschreibung kritischer Punkte und Schwierigkeiten; u. U. nicht gelöste Probleme
 - gewonnene Erkenntnisse
 - Ausblick: Was könnte verbessert/optimiert werden, welche Funktionen fehlen

Fragen? Probleme?

... ich helfe gerne: in meiner Sprechstunde, per Zoom, Discord oder Mail, siehe

<https://www.thm.de/mnd/simon-frank>