**Projekte IT23/TF11 für die Site bapfit.de**

* Alle Projekte sind **Webprojekte** (HTML, CSS, Javascript, PHP, Bootstrap, …)
* Die Projekte können **einzeln oder als Team** bearbeitet werden.
  + Ein Team besteht aus **maximal 4 Schülern** (in einer )
* Alle Projekte müssen **online (bapfit.de) funktionieren und vorgestellt** werden.
  + Sie erhalten dazu einen **ftp-Zugang** eine **Mailadresse** und eine **DB**
* Alle Projekte werden in einem entsprechendem **Unterordner unter bapfit.de** abgelegt
  + erreichbar dann z.B. mit https://bapfit.de/Projekt1
* Alle Projekte werden mit **git** verwaltet
  + und **laufend**! auf Github gespeichert und ich werde als "collaborator" eingeladen.
* Alle Projekte bedürfen auch einer **Planung** und **Dokumentation** (siehe Fach **MDS**).
* **Planung und Doku gehört auch auf github!**
* Bei einem gemeinsamen Projekt braucht **jeder** einen **größeren Programmieranteil**.
* Dazu gehört **PHP und JS und nicht! HTML und CSS-Code**. Das gehört zum Webdesign!
* Bei der **Aufteilung der Projekte ist dies schon zu berücksichtigen**.
* Alle Projekte haben als Basis ein **Loginsystem mit Registrierung:**
  + Nur **Personen mit Schulmail-Account** werden überhaupt (über diese) registriert.
  + @bappassau.de bzw. @bapfit.de
  + Angabe Name (frei) mit **eigener BAP-Mail** und Passwort (mit Richtlinien)
  + Dann Mail zurück für **Bestätigung** (mit z.B. generierter ID)
  + Erst nach der Bestätigung Anmeldung möglich
  + evtl. Abfrage von Zusatzinfo z.B. Gruppe/Klasse it22, it23, it24, ...
* **Onlinetest muss für mich möglich sein**:
* Ich registriere mich, bestätige und logge mich dann ein. Probiere dann aus.
* Dies gilt auch für die Vorführung!
* Am Ende Vorführung/Präsentation des Projektes (**Note im 2.HJ**)

**Projektideen**

1. **Bestellsysteme** 
   * z.B. Pizza, etc.
2. **Notenverwaltung der Schüler** 
   * incl. aut. Berechnung der Gesamtnote, Schnitt, Warnung bei Gefährdung, ...
   * Beachten Sie die reellen Bedingungen (z.B. AE mehrere Lehrer mit Gewichtung)
   * Anpassung an sich ändernde Bedingungen mit z.B. config-Datei (z.B. Gewichtung)
   * Mindestens alle aktuellen Fälle /Bedingungen sollten abgedeckt sein.
   * Verschlüsselung der Daten! (Noten)
3. **Internes Newssystem** 
   * Ankündigungen der Lehrer aber auch der Schüler, kein Chat!
   * News aus Webseiten extrahieren?
4. **Online-Umfrage**
5. **Klassensprecherwahl**
6. **Internes Meldesystem für Probleme Hard-/Software, Ticketsystem**
7. **Eventsystem**
8. **Quiz**
9. **Notizen**
10. **Schwarzes Brett**
11. **Terminverwaltung**
12. **Stundenplan, digitales Klassenbuch, …**
13. **Essensplan**
14. **Eigene Ideen, aber nur nach Vereinbarung!**

**Vorgehensweise / Zeitplanung**

**Vorgehensmodell: Ich schlage vor „**[**Extreme Programming**](https://de.wikipedia.org/wiki/Extreme_Programming)**“**

**Im 1. Halbjahr**

1. **Projekt auswählen. Team bilden.** 
   * Geben Sie dem Projekt einen vernünftigen **Namen**. (Sollte aufs Projekt hindeuten)
   * Schreiben Sie die **vollen Namen der Teammitglieder** in eine Datei members.txt auf
   * Wählen Sie einen **Ansprechpartner** für mich. (auch in Datei markieren)
   * Erstellen Sie ein **privates Github-Projekt** mit obigem Teamnamen.
   * Fügen sie alle Projektmitarbeiter ein. Laden Sie zusätzlich **mich** ein.
   * Sammeln Sie auch alle Infos und Dokumente in diesem Repo. (jetzt und auch später)
2. **Abgabe: (pro Projektteam 1 Word-Dokument) per Chat an mich** 
   * **mit Projektname, Echtnamen der Beteiligten, Ansprechpartner für mich**
   * **Link zum Github-Projekt (Vorher bitte mich einladen!)**
   * **Bis spätestens 22.11.24**
3. **Projekt planen** (immer alle Dokumente sofort auch auf Github!)
   * Erstelle ein **UML-Anwendungsfalldiagramm** (Use-Case-Diagramm)
     + Incl. Generalisierung, Includes, Extends mit Notizen (für Bedingung)!
   * Erstelle anfängliche **Userstorys** als Exceltabelle (siehe mein Bsp.)
   * Userstories als Basis (Excel-Tabelle, die **fortlaufend gepflegt** wird)
   * Erstelle eine **grobe Aufgabenteilung** auf Basis der Use-Cases bzw. Userstorys
   * Schreibe die Namen der Zuständigen in die entsprechende Spalte der Userstory/s.
   * Verwenden Sie bei zu großen Userstorys "[Story-Splitting](https://produktwerker.de/user-story-splitting/)"!
   * Legen Sie dabei auch die **Excel-Tabelle** auch in Github ab
   * Erstelle ein Wireframe für beschriebene Aktionen, z.B. mit <https://app.diagrams.net/>
   * Erstellen Sie ein **UML-Klassendiagramm**.
   * Erstellen Sie ein **E-R-Diagramm** für die Daten.
   * Diese Dokumente auch mit git versionieren und auf github speichern.
   * Diagramme/Dokumente auch als PDF exportieren und speichern!
4. Abgabe: (pro Projektteam)
   * Diagramme, Userstories (Excel), Use-cases, Wireframe, … als Dokumente per Chat
   * **Bis spätestens 1. Februar 2025**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Im 2. Halbjahr (nach der Zwischenprüfung)**

1. 1. Prototyp GUI mit Bootstrap
2. Weitere UML-Diagramme (Sequenz, Aktivitäts, ...)
3. Möglichst bald Online stellen, um die Zusammenhänge zu testen.
4. **Vorführung Ende 2. HJ (Note).** 
   * **Jeder stellt seinen! Programmieranteil vor.**
   * **Juli 2025**