

Zum Praktischen Projekt der Gruppe U27

| Das Projekt | Zu Abgabe | Abgabeart/Abgabeort |
|--|---|---|
| <p>Im Projekt handelt es sich um drei Basisfunktionalitäten:</p> <ol style="list-style-type: none"> 1. Thema erstellen/editieren 2. Frage erstellen/editieren 3. Quiz spielen | <p>Jede der drei Basisfunktionalitäten soll einzeln und vollständig implementiert und "ausgeliefert" werden, d.h. bevor man mit der nächsten anfängt, muss die aktuelle implementiert werden: GUI, BL und DB!</p> | <p>Ausschließlich auf GitHub.</p> <p>Regelmäßig puschen!</p> |
| <p style="text-align: center;">Das Exelblatt "Projektmanagement" (hier im Dokument) als PDF oder Screenshot täglich an Tikko via Teams!</p> | | |
| Vorgehensweise | Zu Abgabe | Abgabeart/Abgabeort |
| Use Case Diagramm in UML erstellen | <p>Use Cases sollen ab jetzt ausschließlich von Hand gezeichnet werden (auf Papier mit Stift)</p> | <p>Als pdf, png oder jpeg von einem Scan oder einem Photo, auf GitHub im Projekt-Folder. (Smart phones können inzwischen auch scanen)</p> |
| Jeden Use Case detailliert beschreiben | Z.B. in einem Word-Dokument | Als PDF auf GitHub im Projekt-Folder |
| Klassendiagramm pro Schicht - GUI + BL + Persistence erstellen (es kann sein, dass BL und Persistence nur modifiziert, wenn überhaupt, werden müssen) | <p>Klassendiagramme sollen ab jetzt ausschließlich von Hand gezeichnet werden. Sie müssen nicht alle Details beinhalten, nur die Eigenschaften und die Methoden der Klasse, die unmittelbar einer Funktionalen Anforderung entspricht.</p> | <p>Als pdf, png oder jpeg von einem Scan oder einem Photo auf GitHub im entsprechenden Package des Projektes.</p> |
| Implementierung inkl. Java Doc | Source Code ohne Java Doc wird als null-Abgabe ausgewertet! | GitHub |

Projekt [Quiz-Game] ([https://github.com/GDayIT/Quiz-Game])

Projektmanagement

| Use Case | Task* | Datum* | Zeitaufwand |
|----------|--|---------------|-------------|
| | | | 197,5 |
| | Party Projekt abschliessen! | Mo. 21 Juli | 8 |
| | Analyse der Mockups anhand Kundenanforderungen | Di. 22 Juli | 1 |
| | Vorbereitung erster Ansätze für die GUI | Di. 22 Juli | 6,5 |
| | Mit den anderen vorgehen besprochen | Di. 22 Juli | 0,5 |
| | Use Case mit/in UML erstellen | Mi. 23 Juli | 2,5 |
| | Storyboards | Mi. 23 Juli | 3 |
| | GUI-Prototyp | Mi. 23 Juli | 3 |
| | GUI-Prototyp | Do. 24 Juli | 2 |
| | 10:00 Teammeeting zu TODO 1 vom Mi. 23.07.2025 | Do. 24 Juli | 0,5 |
| | GUI-Prototyp + Business-Logik-Schicht | Do. 24 Juli | 5,5 |
| | Business-Logik-Schicht + Fake-Daten (Mit Vorbehalt) | Fr. 25 Juli | 0,5 |
| | Fake-Daten implementieren | Fr. 25 Juli | 1 |
| | Business-Logik-Schicht zu Fake-Daten | Fr. 25 Juli | 6,5 |
| | | Sa. 26 Juli | |
| | | So. 27 Juli | |
| | Prototypen ersetzen mit zwei Panels | Mo. 28 Juli | 0,15 |
| | Quizfragen-Panel erstellen (GUI) + logik | Mo. 28 Juli | 5 |
| | Quizlisten-Panel erstellen (GUI) + logik | Mo. 28 Juli | 5 |
| | Fortsetzen der gestrigen Task (zeit auslastung (zt)) | Di. 29 Juli | 0 |
| | Heirat der beiden Panel ohne logik | Di. 29 Juli | 1 |
| | Businesslogik für Quizforming-Panel angepasst | Di. 29 Juli | 2,35 |
| | Panel via delegat + Klassen zuweisung verbinden | Mi. 30 Juli | 15 |
| | Funktionalität Panel sichern, via Delegate/Lambda (zt) | Do. 31 Juli | 1 |
| | Funkt.- P.- sichern,via Serailisierung/Persistenz (zt) | Fr. 01 August | 10 |
| | | Sa. 02 August | |
| | | So. 03 August | |
| | Use Cases zu UML-Diagramm | Mo. 04 August | 2 |
| | Storyboard (Tab/Packages | Mo. 04 August | 1 |
| | Qualitätssicherung einhalten und sich durchlesen | Mo. 04 August | 1 |
| | Panel via delegat + Klassen zuweisung verbinden (zt) | Mo. 04 August | 4 |
| | Funktionalität Panel sichern, via Delegate/Lambda (zt) | Di. 05 August | 2 |

[illegible]

| re.git]) | |
|-------------|---------------|
| | |
| Status | Zeitverbrauch |
| | 196,5 |
| Final | 8 |
| Final | 1 |
| Final | 6,5 |
| Final | 0,5 |
| Final | 2,5 |
| Final | 3 |
| Final | 3 |
| Final | 2 |
| Final | 0,5 |
| Final | 5,5 |
| Final | 0,5 |
| Final | 1 |
| Final | 6,5 |
| Done | |
| Done | |
| Final | 0,15 |
| Final | 10 |
| Final | 4 |
| Final | 0 |
| Final | 1 |
| Final | 2,35 |
| In Proccess | 30 |
| In Proccess | 0 |
| In Proccess | 0 |
| Done | |
| Done | |
| Final | 2 |
| Final | 1 |
| Final | 1 |
| In Proccess | 0 |
| In Proccess | 2 |



| | |
|-------------|-----|
| In Proccess | 2 |
| Final | 2 |
| Final | 3 |
| Final | 2,5 |
| Error | 8 |
| Error | 8 |
| Error | 8 |
| Done | |
| Done | |
| In Proccess | 8 |
| In Proccess | 8 |
| In Proccess | 8 |
| In Proccess | 8 |
| In Proccess | 8 |
| In Proccess | 7 |
| In Proccess | 3 |
| In Proccess | 1 |
| In Proccess | 4 |
| In Proccess | 2 |
| In Proccess | 3 |
| In Proccess | 1 |
| In Proccess | 8 |
| TODO | |
| TODO | |
| TODO | |
| TODO | |
| TODO | |
| TODO | |
| TODO | |
| TODO | |

Notiz:

Da mein Projekt durch Veränderung der Struktur und modularerweise eine bessere Übersichtlichkeit erhalten sind die nötigen Inhalte noch nicht erstellt und noch Progress/Proccess oder auf error gestellt.

Da dies dazu beiträgt nicht nur Programmierer zu sein zu lernen ein Architekt zu werden, Sorge ich nun für nach best practice und konvention mir mitgegebene angemessene Struktur.

d der

ält,

in

in sondern

den

n und

| |
|-------------|
| Status |
| TODO |
| In Proccess |
| Done |
| Forgotten |
| Error |
| Final |

| |
|--------------------|
| Zeit zu Verfügung: |
| Tag |
| Mo. 21 Juli |
| Di. 22 Juli |
| Mi. 23 Juli |
| Do. 24 Juli |
| Fr. 25 Juli |
| Sa. 26 Juli |
| So. 27 Juli |
| Mo. 28 Juli |
| Di. 29 Juli |
| Mi. 30 Juli |
| Do. 31 Juli |
| Fr. 01 August |
| Sa. 02 August |
| So. 03 August |
| Mo. 04 August |
| Di. 05 August |
| Mi. 06 August |
| Do. 07 August |
| Fr. 08 August |
| Sa. 09 August |
| So. 10 August |
| Mo. 11 August |
| Di. 12 August |
| Mi. 13 August |
| Do. 14 August |
| Fr. 15 August |

| |
|------------------|
| Sa. 16 August |
| So. 17 August |
| Mo. 18 August |
| Di. 19 August |
| Mi. 20 August |
| Do. 21 August |
| Fr. 22 August |
| Sa. 23 August |
| So. 24 August |
| Mo. 25 August |
| Di. 26 August |
| Mi. 27 August |
| Do. 28 August |
| Fr. 29 August |
| Sa. 30 August |
| So. 31 August |
| Mo. 01 September |
| Di. 02 September |
| Mi. 03 September |
| Do. 04 September |
| Fr. 05 September |
| Sa. 06 September |
| So. 07 September |
| Mo. 08 September |
| Di. 09 September |
| Mi. 10 September |
| Do. 11 September |
| Fr. 12 September |
| Sa. 13 September |
| So. 14 September |
| Mo. 15 September |
| Di. 16 September |
| Mi. 17 September |
| Do. 18 September |
| Fr. 19 September |
| Sa. 20 September |
| So. 21 September |
| Mo. 22 September |
| Di. 23 September |
| Mi. 24 September |
| Do. 25 September |
| Fr. 26 September |
| Sa. 27 September |
| So. 28 September |

| Info |
|---|
| |
| Task begonnen |
| Soweit erledigt (ergänzungen nötig(fehler frei)) |
| Task wurde vergessen/übersehen |
| (Arbeit Task to Task) Fehler aufgetreten (Harmoniert nicht) |
| Auslieferungsfertig |

| |
|---|
| Ansatz: Plan erfassung |
| KW 30: Alle begonnenen Arbeiten im Party Projekt abschließen. |
| Kundenanforderungen anhand der Mockup zum bevorstehenden praktischen Projekt analy.- |
| Erstelle Erste: Use Case in UML + Storyboards + GUI-Prototyps (Deadline Do. 24.07 10:00.) |
| 10:00 Teammeeting zu TODO 1 vom Mi. 23.07.2025. |
| Mit Vorbehalt: Business-Logik-Schicht + Fake-Daten |
| |
| |
| KW 31: Erstellen von zwei GUI-Panel welche zusammen Quifragen-Panel ergibt + (Sprint 0) |
| Die zwei Panel via delegate und Klassen zuweisung zusammen setzen. (Sprint 1) |
| Funktionalität Panel sichern, via Serailisierung/Persistenz, Delegate/Lambda. (Sprint 2) |
| Sprint 2 fortsetzen |
| Sprint 2 fortsetzen |
| |
| |
| Use Cases zu UML-Diagramm erstellen + Storyboard (Tab/Packages) + Qualitätssicherung |
| Klassendiagramm zu UML erstellen (Tab/Packages) + (Klassen, Felder, Meth.- mit Java Doc) |
| (ERD zu Klassendiagramm) Erstellen |
| Korrekturen des jetzigen Projektstandes durchführen(nach Th-UC + Java Doc + Storyboard) |
| Korrekturen des jetzigen Projektstandes durchführen(nach Th-UC + Java Doc + Storyboard) |
| |
| |
| Korrekturen durchführen(nach Th-UC + Java Doc + Storyboard + Funktionalität) |
| Korrekturen durchführen(nach Th-UC + Java Doc + Storyboard + Funktionalität) |
| Korrekturen durchführen(nach Th-UC + Java Doc + Storyboard + Funktionalität) |
| Thema Panel erstellen incl. (Th-UC + Java Doc + Storyboard) und dem Projekt hinzufügen. |
| Quiz spielen: Datenmodell für die DB erstellen + Java-Prg. für die DB + GUI mit der KI |

Planungen Abschliesen um fortsetzen zu können!

Planungen Abschliesen um fortsetzen zu können!

[illegible]