# Rahmenbedingungen Modul 223 als Probe-IPA

## Fachthema

Eine objektorientierte Multi-User-Applikation wird geplant, umgesetzt, getestet und dokumentiert.

Die objektorientierte Applikation erfüllt folgende Kriterien:

* Front- und Backend
* Zentrale Datenbank
* Mehrere User greifen gleichzeitig auf den gleichen Datenbestand zu
* Zentrale Benutzer- und Rechteverwaltung

## Aufgabenstellung

* Erstellt in der Freizeit (gilt nicht als Arbeitszeit) eine Aufgabenstellung mit einem Umfang von 2-4 A4-Seiten.
* Ein (Teil-)Projekt von der Abteilung und das Arbeiten auf Firmen-Infrastruktur (Notebook, Entwicklungsumgebung, Server) ist möglich, aber nicht obligatorisch (bei der Nutzung von Firmeninfrastruktur, die Erlaubnis von der Abteilung einholen).
* Die Wahl der Technologie ist euch überlassen (Rahmenbedingungen oben müssen aber immer eingehalten werden, => Empfehlung: Technologie von richtiger IPA verwenden).
* Beispiel für eine solche Aufgabenstellung findet ihr auf der Webseite der PK19 in Zürich [www.pk19.ch](http://www.pk19.ch) (<https://pk19.ch/wp-content/uploads/2020/12/Aufgabenstellung-API-Beispiel1.pdf>)
* Das gesamte Projektresultat (Programmcode, Dokumentation und Präsentation) wird am Schluss abgegeben und bei uns archiviert.

## Termine und Zeiten

* Das praktische Projekt dauert 6 Tage (Kurstage) und die täglichen Arbeitszeiten sind von 08:15h-11:30h und 12:30h-16:15h (7h/Tag). Der Abgabetermin des Projektes ist am 6. und letzten Tag um 11:30h (Abgabe Dokumentation).
* Der Umfang des Projektes sind 5x6h plus 3h am Abgabetag (Total 33h). Dazu kommt in den ersten fünf 1h Theorie/Tag und die Abschlussarbeiten am Nachmittag des sechsten Tages von rund 4h.
* Als Vorbereitung bzw. Nachbearbeitung rechnet mit 6h für die Erstellung Aufgabenstellung, der Präsentation und der Demonstration.
* Arbeite täglich rund 60% an der Entwicklung und Umsetzung des Projektes und rund 40% an der Dokumentation des Projektes.
* Die Vorbereitung/Erstellung der Aufgabenstellung, der Präsentation und der Demonstration werden in der Freizeit realisiert und zählen nicht zur Arbeitszeit.

# Aufgabenstellung

## Kandidat:in

|  |  |  |  |
| --- | --- | --- | --- |
| *Name* | Dainese | *Vorname* | Jordan |
| *Fachrichtung* | Applikationsentwickler | *Lehrgang:* | 2020 |
| *Telefon* | 0788461103 | *Mail:* | Jad\_2014@icloud.com |

## Titel

## Terminagenda

## Beschreibung

## Eine Weblösung für eine persönliche Agenda, bei der man Termine hinzufügen, editieren und löschen kann. Ein User soll sich anmelden können, um auf seine Einträge zugreifen zu können.

## Ausgangslage

## Im Modul M223 führen wir eine Mini-IPA durch, um uns auf die richtige vorzubereiten. Dadurch

## erhalten wir Praxiserfahrung und Wissen in welche Richtung es bei der richtigen IPA gehen wird.

## Bei all den Terminen, die man heutzutage hat, ist es manchmal schwer, den Überblick zu behalten. Dies versuche ich zu vereinfachen. Mit der personalisierten Agenda kann man einfach den

## Überblick behalten, sich neue Termine einfach eintragen und, falls nötig, auch wieder entfernen.

## Dieses Tool werde ich al seine Weblösung neu erstellen, um einen leichten Zugriff zu ermöglichen.

## Detaillierte Aufgabenstellung

## Bei der Aufgabe soll eine Webapplikation erstellt werden, die mit einem ergonomischen Nutzerinterface, einem Backend für Datenverarbeitungen und einer Datenbank, um die Daten zu speichern, ausgestattet sein soll.

## Ein Nutzer kann sich registrieren und/oder Anmelden um mit seiner persönlichen Agenda zu interagieren.

## Akzeptanzkriterien

## - Realisierung der Applikation

## Akzeptanzkriterien

## Realisierung der Applikation

## Saubere Dokumentation

## - Pünktliche Abgabe gemäss definiertem Datum

## Nicht-Funktionale Anforderungen

## Die Applikation soll leicht zu bedienen sein

## Der Code ist strukturiert und kommentiert

## Funktionale Anforderungen

## Ein Benutzer kann sich anmelden

## Ein Benutzer kann sich registrieren

## Ein Benutzer kann eigene Termine erstellen

## Ein Benutzer kann eigene Termine editieren

## Ein Benutzer kann eigene Termine löschen

* Ein Benutzer hat eine Übersicht über all seine Termine
* Erstellte Termine werden in einer Datenbank gespeichert
* Beim wieder Einloggen eines Benutzers werden seine Termine aus der Datenbank geladen
* Ein Adminbenutzer hat Zugriff auf ein Adminpanel
* Mit dem Adminpanel kann ein Admin Nutzer bearbeiten (erstellen, löschen, editieren)
* Ein Admin kann weitere Adminbenutzer erstellen

## Individuelle Bewertungskriterien

**Drei** individuellen Kriterien aus Kriterienkatalog, ohne Mehrfachbewertungen, wählen (in der effektiven IPA werden **sieben** individuelle Kriterien gewählt).

1. Individuelles Bewertungskriterium

|  |  |
| --- | --- |
| *Nummer Katalog-Kriterium - Bezeichnung* | |
| 166 - Codingstyle - lesbarer Code | |
| *Definition (Leitfrage)* | |
| Ist der Code lesbar geschrieben, gut gegliedert und ist die Namensgebung gut gewählt? | |
| *Gütestufe 3* | *Gütestufe 2* |
| Die Namensgebung entspricht den Vorgaben oder ist einfach gut gewählt. Die Struktur des Codes ist ebenfalls gemäss möglicher Richtlinien oder einfach übersichtlich gemacht. Es ist eine gewisse Einheit zu sehen in der Art und Weise, wie der Code strukturiert ist (d.h. es ist überall etwa gleich gemacht). | Die Namensgebung ist ab und zu ungeschickt gewählt, Vorgaben sind teilweise berücksichtigt. Die Codestruktur ist uneinheitlich (so dass Lesbarkeit leidet). |
| *Gütestufe 1* | *Gütestufe 0* |
| Die Namensgebung ist öfters verwirrend oder unpräzise. Dem Code fehlt es an einigen Stellen an klarer Struktur. Vorgaben sind nicht berücksichtigt. | Die Namensgebung ist verwirrend oder unpräzise. Der Code ist schlecht lesbar. |

1. Individuelles Bewertungskriterium

|  |  |
| --- | --- |
| *Bezeichnung* | |
| 121 - Software-Ergonomie | |
| *Definition (Leitfrage)* | |
| Ist die nötige Benutzerfreundlichkeit/SW-Ergonomie implementiert? | |
| *Gütestufe 3* | *Gütestufe 2* |
| GUI-Elemente sind  -der Funktion entsprechend gewählt  -verständlich beschriftet  -sinnvoll gruppiert und  -korrekt angewendet. | Alle Bewertungspunkte sind weitgehend erfüllt. |
| *Gütestufe 1* | *Gütestufe 0* |
| Es sind bei jedem Bewertungspunkt nur Ansätze vorhanden, oder: es sind ein bis zwei Bewertungspunkte gar nicht erkennbar. | Die Benutzerfreundlichkeit ist mangelhaft. |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| *Bezeichnung* | |
| 194 - Plausibilisierung der Benutzer-Eingaben | |
| *Definition (Leitfrage)* | |
| Werden die Eingaben des Benutzers überprüft? | |
| *Gütestufe 3* | *Gütestufe 2* |
| Alle Eingabefelder werden überprüft. Es ist eindeutig gekennzeichnet, welche Felder Pflichtfelder sind. Für den Benutzer ist ersichtlich, welche Wertebereiche zulässig sind. Findet die Plausibilisierung eine Fehleingabe, so wird der Benutzer mit konkreten Hinweisen geführt, das entsprechende Feld wird aktiviert. | Plausibilisierung findet statt, Feedback an Benutzer ist mangelhaft/nicht eindeutig/unvollständig. Nur korrekte Daten werden übermittelt. |
| *Gütestufe 1* | *Gütestufe 0* |
| Eingaben werden plausibilisiert, aber bei Fehlern oder fehlenden Eingaben sind die bisher gemachten Eingaben verloren oder die fehlerhaften Eingaben werden trotzdem übermittelt.Oder: es werden nicht alle Eingaben ueberprueft, welche ueberprueft werden sollten. | Es findet keine Plausibilisierung statt. |

1. Individuelles Bewertungskriterium

## Mittel und Methoden

## Das Modul findet hauptsächlich in der Berufsbildung der Siemens statt. Ich werde meinen eigenen Laptop verwenden. Ich entwickle mit folgenden Programmen: Visual Studio Code, Visual

## Studio und Versionierung mit GitHub.

## Vorkenntnisse

- Backend + API erstellt mit .NET Core

- Objektorientierte Applikationen entwickelt

## Vorarbeiten

## - Aufgabenstellung wurde erstellt

## - Informationen und Fragen über den Ablauf wurden geklärt und ausgetauscht

## Neue Lerninhalte

## - Ergonomisches UI entwickeln

## - Vollständige Applikation mit Frontend, Backend und Datenbank alleine entwickeln

## Arbeiten in den letzten 6 Monaten

## - Tunneltool mit JavaScript

## - Einführung Cmake und C++

## - API erstellt mit C# (.NET Core)

## - Automatisierte Tests schreiben

## Tage, an welchen an der Facharbeit gearbeitet wird

14.-15. + 17. + 21.-22. + 24. März 2023(5 ½ Tage – Abgabe 24. März 2023 um 11:30h)

## Bemerkungen

Die Rückmeldung findet am individuellen Präsentationstermin statt. Sie findet nur mündlich statt und es ist in den Interessen des Kandidaten Notizen zu machen. Die Noten werden erst nach Abschluss aller Arbeiten verschickt.