

HANDLUNGSEMPFEHLUNG: NUTZERORIENTIERTE SOFTWAREENTWICKLUNG IM PROJEKTUNTERRICHT

User Centered Design als Methode zur Entwicklung
eines Softwareproduktes im Unterricht

Prof. Dr. Sven Bertel
Lena Beck
Natalia Buchmann
Johann Kirchkesner
Vanessa Schomakers



in Kooperation mit der
Auguste-Viktoria-Schule Flensburg



sowie dem Institut für Qualitäts-
entwicklung Schleswig-Holstein

Bernd Clausen
Timo Räker

Philipp Möller

Dieses Dokument wurde erstellt in enger Zusammenarbeit mit der Auguste-Viktoria-Schule Flensburg und dem Institut für Qualitätsentwicklung an Schulen Schleswig-Holstein.

Die Handlungsempfehlung wurde entwickelt im Zuge des Projektes „Kompetenzen für eine digitale Welt – Partizipative Entwicklung von Apps zum Training räumlicher Apps. Ein schülerinnen- und lehrerinnenorientiertes Good Practice für Bildung in der digitalen Welt“, durchgeführt an der Hochschule Flensburg am Center for Interaction, Visualization and Usability (CIVU) unter der Leitung von Prof. Dr. Sven Bertel.

Herausgegeben im Januar 2022, Flensburg

Kontakt:

Prof. Dr. Sven Bertel

sven.bertel@hs-flensburg.de

Lena Marie Beck

lena-marie.beck@hs-flensburg.de

Info

Handlungsempfehlung für den Informatik-unterricht in der gymnasialen Oberstufe

Angepasst an die aktuellen Fachanforderungen Informatik Schleswig-Holstein 2021

Laufzeit: ein Schuljahr projektbezogener Unterricht zum Thema Softwareentwicklung

- Entwicklung einer App/Webseite auf Basis von Nutzeranforderungen, Design-Konzepten, Prototypen und Nutzertests Modular anpassbar an Stundenumfang und Vorwissen

Inhaltsverzeichnis

<i>Inhaltsverzeichnis</i>	2
1. Abstract	3
2. Das Projekt	3
2.1. Informatik: Begleitung des Schulunterrichts	3
2.2. Chemie: Die Lern-App	4
2.3. Entwicklungen in der Bildungslandschaft	5
2.4. Infos zum Einstieg	7
2.4.1. User Interface (UI) Programmierung	7
2.4.2. Interfacedesign/User Experience (UX).....	7
2.4.3. Bezug zu den Fachanforderungen.....	8
3. Resultierende Handlungsempfehlung: Die Modulbausteine	9
3.1. Vorbedingungen für diese Handlungsempfehlung:	9
3.2. Übersicht und Modulaufbau	10
3.3. MB01: das Kernmodul UCD	12
3.4. MB02: Programmierung: Start-Modul	15
3.5. MB03: Design: Start-Modul	17
3.6. MB04: Papier-Prototypen	19
3.7. MB05: Usability-Testing: Papierprototypen	20
3.8. MB06: Programmierung: Grundlagen-Modul	21
3.9. MB07: Design: Grundlagen-Modul	22
3.10. MB08: Klick-Prototypen	24
3.12. MB09: Usability-Testing: Klickprototypen	26
3.13. MB10: Programmierung: Vertiefungs-Modul	28
3.14. MB11: Design: Vertiefungsmodul	29
3.15. MB12: Produkt: Nativer Programmier-Prototyp	30
3.16. MB13: Usability-Testing: Produkt: Nativer Prototyp	31
3.17. MB14: Abschlussmodul: Projektabschluss/ Ergebnispräsentation	32
3.18. Ausgewählte Beispiele für die Kombination der Modulbausteine	33
4. Verknüpfung mit den Kompetenzen in den Fachanforderungen	36
5. Literaturverzeichnis	42
6. Graphikverzeichnis	43
7. Tabellenverzeichnis	43

1. Abstract

Die folgende Handlungsempfehlung stellt eine Handlungsempfehlung für den Informatikunterricht dar und ist auf die gymnasiale Oberstufe ausgelegt, lässt sich jedoch durch die modulare Gestaltung ebenfalls anpassen an andere Schulformen sowie Klassenstufen. Das Konzept ist in kleinere Modulbausteine unterteilt, die sich – angepasst an die Faktoren Vorwissen der Schülerinnen und Schüler und die verfügbare Stundenzahl pro Woche – kombinieren lassen. Dadurch lässt sich diese Handlungsempfehlung sowohl für ein Wahlpflichtfach Informatik (2-stündig) als auch für den Unterricht in einer Informatik-Profiloberstufe (4-stündig) entsprechend anpassen und bei vollem Umfang auf ein ganzes Schuljahr ausdehnen (38-42 Schulwochen).

2. Das Projekt

Um die folgende Handlungsempfehlung etwas besser einordnen zu können wird hier kurz das Projekt „Kompetenzen für eine digitale Welt“ vorgestellt, welches als Pilotprojekt für die Handlungsempfehlung dient.

Das Projekt wurde im Oktober 2019 gestartet und läuft bis Oktober 2022 an der Hochschule Flensburg. Das Projektteam ist interdisziplinär aufgestellt und besteht aus einem Informatiker, zwei Interface-Designerinnen, einer Pädagogin und dem Projektleiter, welcher die Professur für Usability inne hat.

Das Projekt verläuft in zwei parallelen Strängen, welche in den folgenden Abschnitten beschrieben werden.

2.1. **Informatik: Begleitung des Schulunterrichts**

In der Zeit von Januar 2020 bis Juni 2021 wurde ein Wahlkurs Informatik in der Oberstufe (10./11.Klassenstufe) am Flensburger Gymnasium Auguste-Viktoria-Schule (AVS) begleitet und gemeinsam mit den Schülerinnen und Schülern der Entwicklungsprozess einer App anhand des User Centered Design-Ansatzes durchlaufen. Dadurch bekamen die Schülerinnen und Schüler Kernkompetenzen im Bereich Softwareentwicklung und sammelten Erkenntnisse über die Entwicklung und Programmierung einer App, aber auch das Verständnis für die vorher ablaufenden Prozesse wie Nutzeranforderungen sammeln, Prototypen entwickeln, Designgrundlagen als auch Usability Testing. Diese sind Grundlage für ein nutzerfreundliches Produkt, was in diesem Fall eine App für den Chemieunterricht Klassenstufe 9 war. Zielgruppe des zu entwickelnden Produktes waren somit Schüler und Schülerinnen der Mittelstufe. Gearbeitet wurde in diesem Projekt iOS-basiert (XCode, Swift) aufgrund der technischen Ausstattung der AVS, ist aber genauso auf einer Android-Plattform umsetzbar.

Die enge Zusammenarbeit mit der AVS und dem Institut für Qualitätsentwicklung an Schulen Schleswig-Holstein (IQSH) waren für die Ausarbeitung dieser Handlungsempfehlung – auch

auf Basis der aktuellen Fachanforderungen Informatik – maßgeblich für den Erfolg des Projektes.

2.2. Chemie: Die Lern-App

Im Zuge des Projektes wird parallel zur Entwicklung im Schulunterricht durch das Projektteam der Hochschule eine App gestaltet und programmiert. Diese App dient der Unterrichtsergänzung zum Training räumlicher Fähigkeiten, hier mit dem Fokus auf den Molekülbau im Chemieunterricht. Der digitale Molekülbaukasten soll eine digitale Ergänzung des Unterrichts und eine Erweiterung des Chemielabors sein.

Die Produktvision:

„Unsere Lösung wird Lehrerinnen und Lehrern (Zielgruppe) dabei helfen, größere Lernerfolge im Chemie-Unterricht bei den SuS (Ziele) zu erreichen, in dem es ihnen digitale Unterstützung bei Aufgaben wie Moleküle zusammenbauen, transformieren und vergleichen in verschiedenen Repräsentationen (2D/3D) (Aktivitäten) ermöglicht ohne die Einschränkung und den Zeitverlust durch aufwendige Steckkästen (Probleme).

Im Gegensatz zu Steckkästen und den meisten bestehenden Apps (Alternativlösungen) kann es Feedback zum Bearbeitungserfolg der Aufgaben an SuS und LuL geben, einfach und schnell eingesetzt werden und erhöht durch gute Didaktisierung, Gestaltung, Implementierung und Usability die Motivation der Schülerinnen und Schüler (Alleinstellungsmerkmal).“

Die entwickelte App soll dann im weiteren Testvorgehen an der Projektpartnerschule eingesetzt werden. Mit der App sollen dann entsprechende Studien zur räumlichen Kognition mit Fokus auf den Molekülbau stattfinden.

2.3. Entwicklungen in der Bildungslandschaft

Dieses Projekt und die daraus resultierende Handlungsempfehlung knüpfen an die Entwicklungen in der schulischen Bildung an. Sowohl auf Länder- als auch auf Bundesebene sind entsprechende Bestrebungen für eine verbesserte Bildung im Bereich digitales Lernen und eine Stärkung der MINT-Fächer zu finden. Bereits 2016 wurde im Strategiepapier der Kultusministerkonferenz festgelegt, dass durch eine entsprechende technische Ausstattung die technische Infrastruktur gestärkt werden soll und „lebenslanges Lernen zu einem unabdingbaren Erfordernis“ (Kultusministerkonferenz (KMK), 2016, S. 21) geworden ist. So sind in den letzten Jahren unter anderem folgende Beschlüsse und Strategiepapiere veröffentlicht worden:

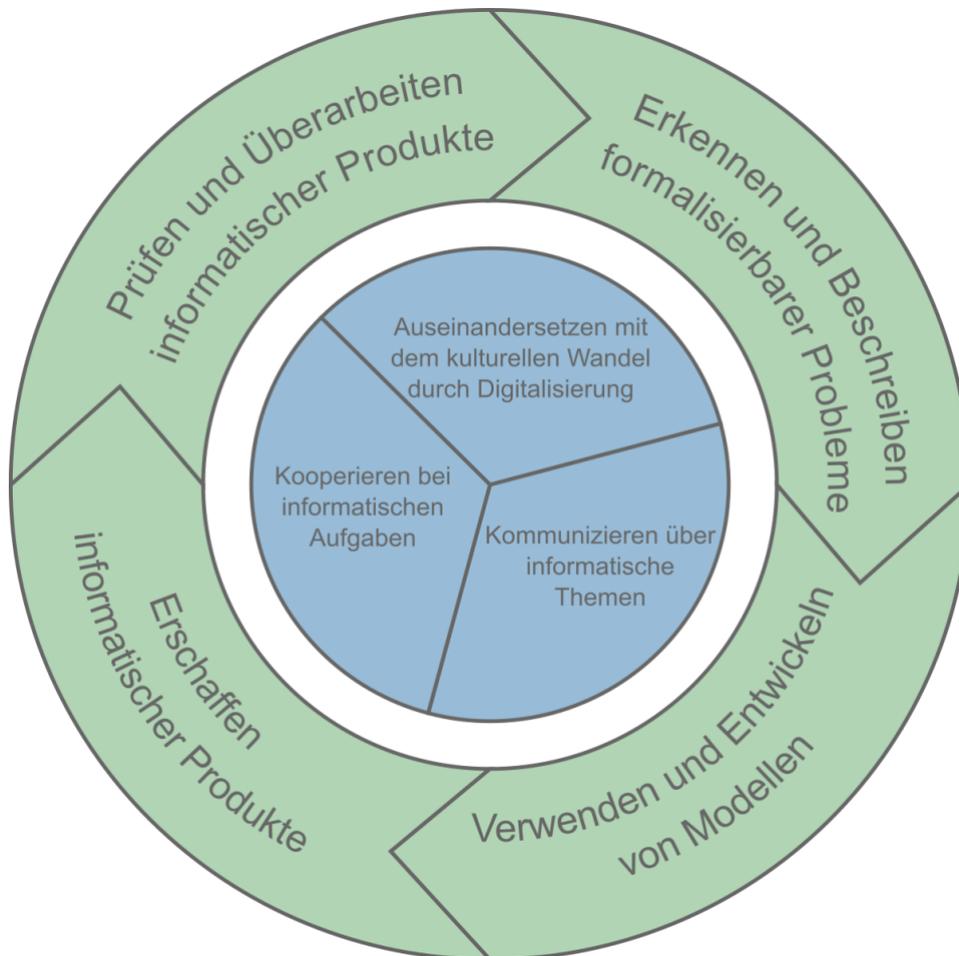
- Fachanforderungen Informatik SH 2021 (Schleswig-Holstein Ministerium für Bildung, Wissenschaft und Kultur, 2020)
- Fachanforderungen Medienkompetenz SH 2018 (Schleswig-Holstein Ministerium für Bildung, Wissenschaft und Kultur, 2018)
- DigitalPakt Schule des Bundes (Bundesministerium für Bildung und Forschung (BMBF), 2019) sowie weitere Ergänzungen 2020 und 2021 (Bundesministerium für Bildung und Forschung (BMBF), 2021)
- KMK Strategiepapier 2016 (Kultusministerkonferenz (KMK), 2016)

Auch auf Länderebene sind viele Entwicklungen zu sehen. So beispielsweise die Digitalisierungsstrategie des Landes Schleswig-Holstein. 2018 wurde eine Ergänzung zu den Fachanforderungen herausgegeben, in der die Anforderungen für die fächerübergreifende Medienkompetenz und das Lernen mit digitalen Medien ausformuliert sind. Für das Fach Informatik gibt es derzeit einerseits in Schleswig-Holstein die Weiterbildungsoffensive, um neue Lehrkräfte für das Fach zu gewinnen, des weiteren wurde Anfang 2021 eine erneuerte Version der Fachanforderungen veröffentlicht. Die neuen Fachanforderungen benennen als Handlungsorientierung eine didaktische Leitlinie unter der das Fach Informatik zukünftig unterrichtet werden soll:

"Nur wenn Schülerinnen und Schüler wiederholt die Gelegenheit bekommen, Handlungsprodukte in Einzel- und Gemeinschaftsarbeit zu erschaffen, können sie die [...] angestrebten Kompetenzen aufbauen. Dabei ist der Kreativität der Schülerinnen und Schüler hinreichend Raum zu geben." (Fachanforderungen Informatik SH, 2021, S.11)

Die Erstellung von Handlungsprodukten findet auch in den Kompetenzbeschreibungen Einzug. So stützen sich die prozessbezogenen Kompetenzen (Abbildung 1, siehe äußerer Kreis) auf das iterative Vorgehen der informatischen Produktentwicklung. Das vorgestellte Projekt folgt den gleichen Prinzipien, so dass in der Umsetzung in allen Phasen die prozessbezogenen Kompetenzen der Fachanforderungen berücksichtigt werden können.

Abbildung 1: Prozessbezogene Kompetenzen (Fachanforderungen Informatik SekI+II, IQSH, S.13)



Mit dem inhaltsbezogenen Kompetenzbereich "Softwareentwicklung" für die Sekundarstufe II wird der Prozess als expliziter Gegenstand des Informatikunterrichts benannt. In dieser Handlungsempfehlung finden sich deshalb immer wieder Verweise auf die Kompetenzen der Fachanforderungen, die die Integration des Projektes in den Schulalltag erleichtern sollen.

Um diese Anforderungen und gewünschten Entwicklungen umsetzen zu können braucht es geeignete Handlungsempfehlungen, Weiterbildungen, Fortbildungen und Lehrkonzepte. Eine Handlungsempfehlung in Bezug auf Softwareentwicklung ist an dieser Stelle ausformuliert.

2.4. Infos zum Einstieg

In diesem Kapitel wird kurz erläutert, welche Rolle und Relevanz die beiden folgenden Bereiche für den User Centered Design-Prozess und das durchgeführte Projekt

2.4.1. User Interface (UI) Programmierung

Die Handlungsempfehlung besteht zu einem nennenswerten Anteil aus Programmieren. In diesem Kontext geht es vorwiegend um die Schnittstelle zwischen Menschen und Programmen, dem User Interface. Dazu werden die Bestandteile einer App, bzw. einer Webseite, in Komponenten zerlegt und erläutert, wie diese in Code umgesetzt werden. Vor Beginn des Projektunterrichts sollten die Schülerinnen und Schülern bereits Erfahrung mit einer Programmiersprache gesammelt haben und diese nutzen können, um in dieser Programmiersprache sowohl das User Interface, als auch die innere Logik der Anwendung umsetzen zu können. Beispielsweise das Herunterladen einer Datei nach Klick eines Buttons und dem Signalisieren des Voranschreitenden und des Abschlusses des Ladevorganges. Im Laufe des Projektes lernen die Schülerinnen und Schüler die Software-Bibliothek und die Werkzeuge zum Erstellen des User-Interfaces anhand von kleinen Anwendungen kennen. Dabei empfehlen sich deklarative Bibliotheken und Sprachen wie HTML oder XML im Gegensatz zu imperativen Sprachen wie reines C oder Java. Zudem sind Komponentenbasierte UI-Bibliotheken wie React, SwiftUI, Jetpack Compose oder Flutter (beinhaltet Inhalt, Darstellung und Logik für sich selbst) leichter zu handhaben, als beispielsweise klassische Webseiten (Trennung von Inhalt, Darstellung und Logik in separaten Dateien).

Zum Schluss sollen die Ergebnisse aus der Konzeption des Interfacedesigns genutzt werden, um die geplante App, bzw. Webseite umzusetzen.

2.4.2. Interfacedesign/User Experience (UX)

Als Interface wird die Oberfläche eines Objektes, welches die Kommunikation bzw. Interaktion zwischen Mensch und Objekt ermöglicht, bezeichnet. Dies beginnt bei alltäglichen Dingen wie einer Türklinke, geht im technischen Sinne weiter mit Knöpfen und Schaltern bis hin zu digitalen Interfaces wie Screens und deren Gestaltung, z.B. klickbare Buttons. Interfacedesign ist somit die Gestaltung dieser Oberfläche. (Usability.gov, 2021) (Digital.gov team in the U.S. General Services Administration (GSA) Technology Transformation Service, 2021)

Das Thema ist für diese Handlungsempfehlung und den Informatikunterricht deshalb von Bedeutung, weil die Entwicklung einer App oder einer Webseite mehr ist als das reine Programmieren und der technischen Implementierung noch einige Schritte vorausgehen. Sinnvolle Funktionen werden mit Blick auf die zukünftige Nutzerinnen und Nutzer - die sogenannte Zielgruppe - ausgewählt, konzipiert und gestaltet.

Im UX-Design wird das Nutzungserlebnis im weiteren Sinne betrachtet. Es werden Fragen bearbeitet, wie z.B.: In welchem Kontext wird das Produkt verwendet? Welche Rahmenbedingungen gibt es? Welche Eigenschaften hat die Zielgruppe, welche Wünsche und Schwierigkeiten gilt es zu berücksichtigen? Durch die Berücksichtigung dieser Aspekte wird das endgültige Produkt durchdachter und passender für den Einsatz durch die Nutzerinnen und Nutzer und es entsteht ein verbessertes Nutzungserlebnis.

2.4.3. Bezug zu den Fachanforderungen



Viele der Module weisen einen expliziten Bezug zu den Fachanforderungen und den dort formulierten Kompetenzen auf. Diese Bezüge sind bei den jeweiligen Modulen in der Graphik (siehe rechts) vermerkt. Außerdem findet sich am Ende des Dokuments eine Auflistung der Module mit den Referenzen (Kapitel 4.1, Tabelle 1) sowie eine Tabelle der gesamten Kompetenzen der Fachanforderungen (Kapitel 4.2, Tabelle 2).

3. Resultierende Handlungsempfehlung: Die Modulbausteine

3.1. Vorbedingungen für diese Handlungsempfehlung:

Allgemeine Grundvoraussetzungen:

- Sowohl die Vorkenntnisse (sowohl der LuL als auch der SuS) und der verfügbare Stundenumfang sind ausschlaggebend für die Zusammensetzung der Modulbausteine. Dafür haben wir die Handlungsempfehlung modular gestaltet, um es an diese beiden Faktoren anpassen zu können.
- Von Vorteil sind bereits erste vorhandene Grundkenntnisse in der (von der Lehrkraft) ausgewählten Programmiersprache. Dies gilt, sofern der Bereich Programmieren (Module 2,6 und 10) sowie die Module "Produkt" beim Prototyping (Modul 12) und Usability Testing (Modul 13) durchgeführt werden sollen. Vorkenntnisse sind somit nicht für jedes Modul zwingend nötig, aber von Vorteil.

Info

Einzelne Module oder Teile davon lassen sich auch in andere Fächer auslagern, um fächerübergreifenden Unterricht zu ermöglichen. So lassen sich beispielsweise die Designthemen Farbenlehre oder Typographie in den Kunstunterricht integrieren.

Grundvoraussetzungen zur Infrastruktur:

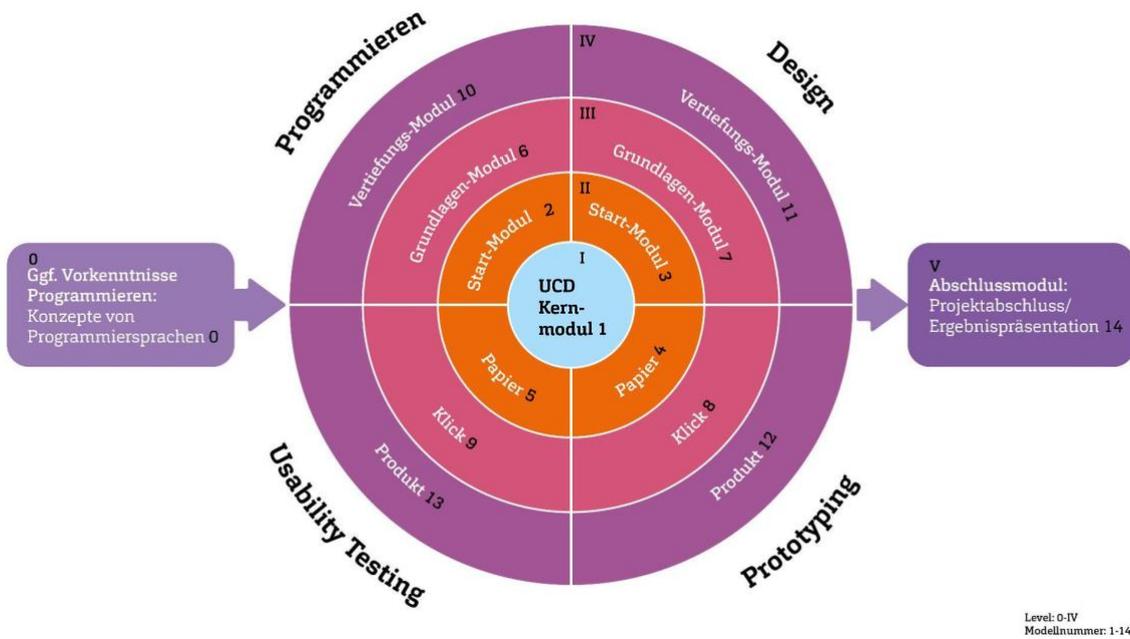
- Ein fester Kurs in Informatik mit 2 bis 4 Stunden Unterricht pro Woche (à 45 Minuten)
- Laufzeit über ein Schuljahr, das entspricht ca. 38-42 Schulwochen.
- Hardware-Ausstattung mit entsprechenden Geräten: Laptops/PCs für die SuS, auf denen die nötigen Programme benutzt werden können. Dies gilt nicht für alle Module, aber für die Bereiche Programmieren (Module 2,6,10), die Module zum Klickprototypen (Module 8, 9) sowie Produkt (Module 12,13) nötig.
- Software-Ausstattung für den Bereich Programmieren (Module 2,6,10) sowie Produkt (Module 12,13) und auch die Module zum Klickprototypen (Module 8, 9). Die entsprechenden Programme zum Coden sowie zur Erstellung des Klickprototypen sind durch die Lehrkraft auszuwählen. Entsprechende Vorschläge sind unten in den jeweiligen Modulen hinterlegt.

3.2. Übersicht und Modulaufbau

- Umfang
 - Hier wird ein grober Zeitrahmen in Form von Stundenanzahl (à 45 Minuten) gegeben, um die Dauer jedes Moduls einschätzen zu können. Dabei wird eine Spanne angegeben, damit Faktoren wie verfügbare Unterrichtsstunden pro Woche und Kenntnisstand von Lehrkraft und Schülerinnen und Schülern berücksichtigt werden können. Der Stundenumfang jedes Moduls ist jedoch lediglich eine Empfehlung und auf die Durchführung eines Projektes über ein ganzes Schuljahr ausgelegt. Die Module lassen sich natürlich auch ganz individuell gestalten und auch nur in Teilen umsetzen.
- Vorbedingungen:
 - Welche Module müssen vorher durchgeführt worden sein?
 - Kurze Beschreibung, was an weiterem Vorwissen, Kompetenzen, Material, Geräten und Zeit da sein sollte.
- Kurzbeschreibung:
 - Was beinhaltet dieser Baustein? Was wird gemacht und worum geht es?
- Ablauf:
 - In welcher Reihenfolge wird was gemacht von wem und wie?
- Ziel:
 - Was wird der erreichte Zustand nach Durchführung des Bausteins sein? Welches Wissen wird vermittelt sein, welche Kompetenzen wurden erworben und welches Produkt ist entstanden?
- Zugehörige Materialien:
 - Beispielsweise Folien, Arbeitsblätter für SuS, Infomaterial für LuL, welche in digitaler Form mitgegeben werden.
- Weitere Quellen zum Modul:
 - Hier sind weitere Quellen und Links zu Webseiten, Graphiken, Büchern etc. zum jeweiligen Modul zu finden.

Zur Veranschaulichung hier eine visuelle Übersicht über die Module:

Abbildung 2: Visuelle Übersicht der Module



Im Folgenden werden die einzelnen Module jeweils ausformuliert und beschrieben.

3.3. MB01: das Kernmodul UCD



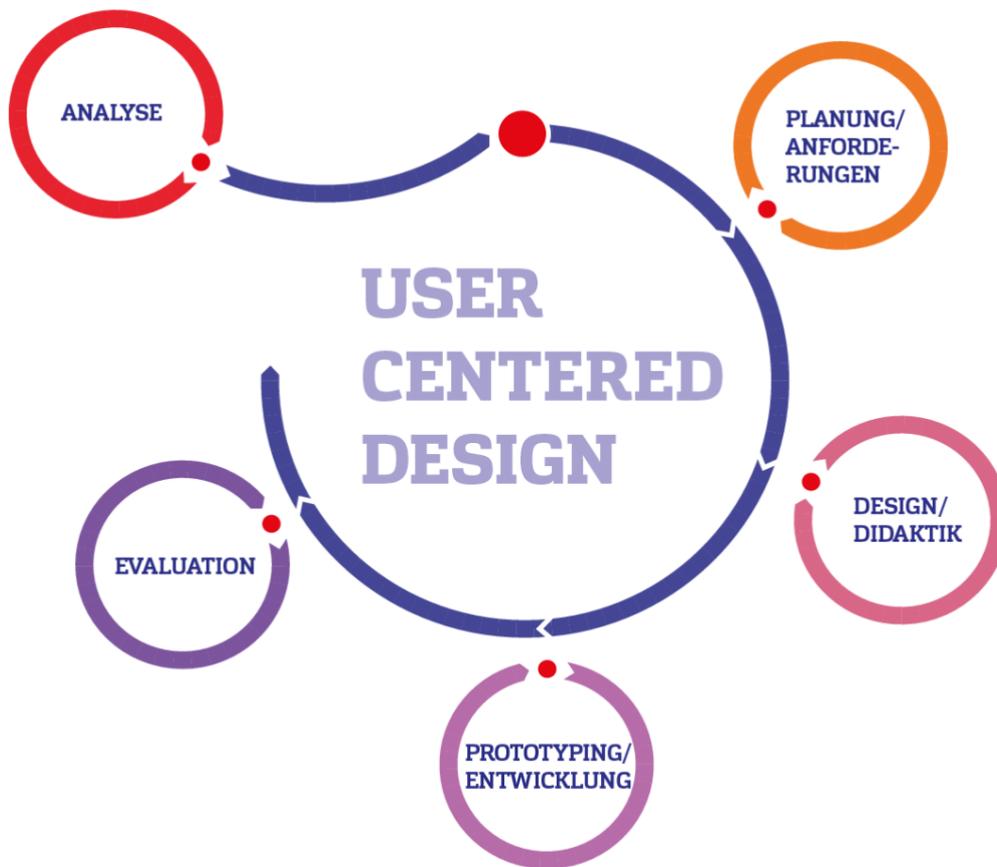
- Umfang:
 - ca. 10-18 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Um die Module aus dem Bereich Programmieren (Module 2,6 und 10) sowie den nativ programmierten Prototypen (Module 12 und 13) durchzuführen sind Vorkenntnisse über die Grundstrukturen im Programmieren nötig. Die Programmierkenntnisse sind für die Umsetzung des gesamten UCD-Prozesses Grundwerkzeug und für die Produkterstellung notwendig.
- Kurzbeschreibung:
 - In diesem Modul werden die grundlegenden Schritte im User Centered Design-Prozess erläutert und der Überblick über das Zusammenspiel der Schritte im Zyklus vermittelt.
 - Anforderungen aus Nutzersicht erheben: Interviews führen ist ein mögliches Werkzeug, um Nutzeranforderungen zu sammeln.
 - Anforderungen Sammeln: User Stories (Schema)
- Ablauf:
 - Zuerst wird ein Überblick über den Zyklus geschaffen , wobei folgende Graphik als Einstieg dient (siehe Abbildung 3)
 - Die einzelnen Etappen werden jeweils kurz vorgestellt und anhand von Beispielen erläutert. Wichtig ist dabei der Aspekt, dass der ganze Entwicklungsprozess als iterativ, also als Zyklus, funktioniert und kein einmaliger Durchgang durch die einzelnen Phasen ist. Das bedeutet, dass ich je nach Intensität immer wieder durch den UCD-Zyklus durchgehe und dabei im Prototypen ein immer konkreteres und ausgearbeiteteres Produkt erhalte. Dadurch wird sich stückweise dem idealen Produkt für die Zielgruppe(n) genähert.
 - Überleitung zum durchzuführenden Projekt: Problemstellung und Ziel werden vorgestellt, z.B. Briefing für eine Chemie-App für die Mittelstufe
 - Zielgruppendefinition: Entweder vorgeben oder gemeinsam entwickeln
 - Anforderungserhebung bei Nutzerinnen und Nutzern, Methode: Interviews mit Personen der Zielgruppe
 - Es werden erste Anforderungen aus Nutzersicht als Grundlage für Designentscheidungen erhoben, indem Interviews geführt werden. Diese müssen mit einem Fragebogen vorbereitet, durchgeführt und ausgewertet werden.
- Ziel:

- Die Schülerinnen und Schüler sollen den iterativen und nutzerorientierten Ansatz des UCD verstehen und im Anschluss in den ausgewählten Modulen anwenden. Der Unterschied zu anderen Entwicklungsprozessen und -vorgehensweisen, wie beispielsweise dem Wasserfallmodell, soll deutlich werden und die Bedeutung der Nutzerinnen und Nutzer für den Entwicklungsprozess im Fokus stehen.
- Erheben von Anforderungen als Grundlage für das Konzept und die weiteren (Design-)Entscheidungen

- Zugehörige Materialien:
 - Graphik UCD-Zyklus
 - Leitfaden Interviewführung
 - Übersichtsgraphik Modulbausteine

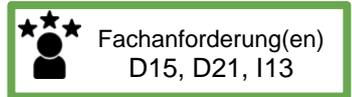
- Weitere Quellen:
 - <https://www.nngroup.com/books/design-everyday-things-revised/>
 - <https://www.interaction-design.org/literature/topics/user-centered-design> (UCD-Prozess, hier auch weitere Links ganz unten zu finden)
 - umfangreiche, kostenlose Infografik zum UCD Prozess: <http://www.paznow.com/ucd/>
 - Überblick Disziplinen im Bereich UCD/UX: <https://github.com/envisprecisely/disciplines-of-ux>
 - zugehörige DIN Normen Überblick: https://de.wikipedia.org/wiki/ISO_9241
 - <https://www.nngroup.com/articles/user-interviews/> (Interviewvorbereitung und Interviewführung)

Abbildung 3: Iterativer Prozess: UCD als Zyklus



Darstellung: Hochschule Flensburg | nach SAP

3.4. MB02: Programmierung: Start-Modul



- Umfang:
 - ca. 10-16 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Die Programmiersprache sollte bereits in vorangegangenen Unterricht kennengelernt worden sein. Die SuS sollten in der Lage sein einfache Anwendungen, wie z.B. Konsolenanwendungen zu erstellen. Die Grundkonzepte der Programmiersprache müssen verstanden worden sein.
- Kurzbeschreibung:
 - Überblick über die Programmierumgebung. (Visual Studio Code, Xcode, Android Studio, etc.)
 - Einführung in die UI-Bibliothek mit Beispielaufgaben zum Kennenlernen der UI-Elemente, dem Stylings von UI-Elemente und einfachen Interaktionen.
 - Erstellung von Oberflächen mit Hilfe von horizontalen und vertikalen Verschachtelungen. (App-Entwicklung)
 - Erstellung von Oberflächen mit Hilfe eines Grid-Layouts. (Web-Entwicklung)
- Ablauf:
 - Vorstellung der Programmierumgebung, der wichtigsten Funktionen und Bereiche.
 - Einführung der ersten UI-Elemente mit zusätzlicher Übung zur graphischen Modifizierung der Farbe, Größe und Abständen.
 - Einführung von vertikalen und horizontalen Gruppierungen von UI-Elementen. (App-Entwicklung)
 - Einführung eines Grid-Layouts von Web-Elementen- (Web-Entwicklung)
 - Idee: Erstellung einer Profalseite anhand eines vorgegebenen Wireframes.
 - Idee: Erstellung einer kleinen App mit Interaktionen, um eine angezeigte Farbe zu ändern.
 - Idee: Oberfläche für existierende Konsolenanwendung (z.B. Taschenrechner)
- Ziel:
 - In diesem Modul soll die Programmierumgebung kennengelernt werden und ein erstes Grundverständnis für den Aufbau einer Oberfläche gewonnen werden. Zudem soll die UI-Bibliothek und ihre Nutzung vermittelt werden.
- Zugehörige Materialien:
 - Video: Deklarative_UI

Info
Sowohl für dieses Modul als auch alle weiteren Module aus dem Programmierbereich gilt folgende Anregung: Unterrichtseinheiten zum Programmieren können auch gebündelt als Projekttag gehalten werden. Dies hat den Vorteil, dass nicht nach jeder Unterrichtseinheit ein Bruch ist, sodass im neuen

- Beispiel Wireframe Profilseite
- Beispiel Musterlösung Profilseite
- Weitere Quellen zum Modul:
 - <https://developer.android.com/jetpack/compose/mental-model#paradigm>
Deklaratives Entwicklungsparadigma
 - <https://thunkable.com/> Thunkable App-Entwicklung mit Visuellen Blöcken wie bei Scratch (Mittelstufe)
 - <https://studio.code.org/projects/applab/new> App Lab App-Entwicklung mit Javascript Blöcken wie bei Scratch (Mittelstufe)
 - <https://developer.apple.com/documentation/swiftui/> SwiftUI natives iOS UI-Framework (Oberstufe)
 - <https://developer.android.com/jetpack/compose> Jetpack Compose natives Android UI-Framework (Oberstufe)
 - <https://flutter.dev/> Flutter Cross-Plattform Framework für Mobile, Web und Desktop (Oberstufe)
 - <https://reactjs.org/> React komponentenbasiertes Javascript Framework (Oberstufe)

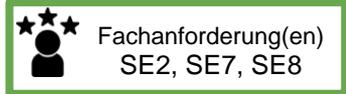
3.5. MB03: Design: Start-Modul



- Umfang:
 - ca. 6-10 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Basismodul UCD (MB01)
- Kurzbeschreibung:
 - Dieses Modul dient dazu, die Bedeutung von Design bei (Software-)Produkten im weiteren Sinne zu vermitteln. Die wichtigsten Begriffe werden besprochen und die Bedeutung dieser Aspekte für Alltag, Produktdesign und Softwareentwicklung erläutert.
 - Klärung Grundbegriffe: Interface, GUI, Interaktionsdesign, User Experience, Konzeption
 - https://de.wikipedia.org/wiki/User_Experience
 - <https://www.interaction-design.org/literature/topics/ui-design>
 - <https://cadabrastudio.medium.com/top-5-best-ux-infographics-4196e47cd22f>
 - Rolle von Design: Enthält mehr als die Optik; auch Interaktionsdesign, Auswahl von Funktionalitäten
 - Erstellen eines Grobkonzepts, Methode: User Flowcharts
 - <https://servicedesigntools.org/tools>
 - <https://www.thisisservicedesigndoing.com/methods>
<https://germanupa.de/arbeitskreise/arbeitskreis-user-research/methodensammlung-user-research>
- Ablauf:
 - Anhand von Folien wird eine theoretische Einführung gegeben
 - Veranschaulichen durch Beispiele für „gutes“ und „schlechtes“ Design; weitere Beispiel-Recherche kann als Aufgabe an die SuS gestellt werden.
 - Die Erkenntnisse aus den Interviews (Kernmodul UCD) fließen in erste Skizzen und User Flowcharts analog mit Stift und Papier oder digital.
- Ziel:
 - Die Relevanz von Design und der nutzungsfreundlichen Gestaltung wird deutlich.
 - Die Abstraktion zu Anforderungen von Zielgruppen wird bewusst.
 - Erste konzeptionelle und gestalterische Ideen werden als User Flowcharts festgehalten und dienen als Grundlage für weitere Entscheidungen.
- Zugehörige Materialien:
 - Beispiele Flowcharts

Info
Das Thema User Flowcharts bietet sich beispielsweise als Klausurersatzthema an. Anregungen z.B. hier: <https://uxqb.org/de/dokumente/>

- Weitere Quellen:
 - Wireframes/Flowchart-Tools (Alternativen):
 - <https://balsamiq.com/wireframes/>
 - <https://www.adobe.com/de/products/xd.html>
 - <https://www.axure.com/edu> (free edu licence)
 - <https://careerfoundry.com/en/blog/ux-design/what-are-user-flows/>
(Erläuterungen zu UserFlowcharts)



3.6. MB04: Papier-Prototypen

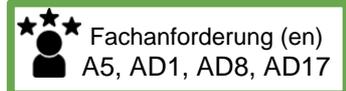
- Umfang:
 - ca. 2-8 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module MB01, MB03
- Kurzbeschreibung:
 - In diesem Modul werden die Schülerinnen und Schüler ihre ersten eigenen Prototypen erstellen. In diesem Durchgang des UCD-Zyklus sind es Papierprototypen, die gebaut werden und später im Modul MB05 getestet werden können.
- Ablauf:
 - Mit Schere, Kleber, Papier und Stiften werden auf Grundlage der User Flowcharts die relevanten Screens mit zugehörigen Elementen erstellt und damit die konzeptionellen Ideen weiterentwickelt und erprobt. Mit den Screens werden die typischen Interaktionen nachgestellt.
 - Grundlage für die Auswahl des Detailgrads und der benötigten Screens sind die Flowcharts; es sollten die wichtigsten ca. 2-4 Abläufe abgebildet werden
 - Als Hilfestellung kann die Frage gestellt werden: Welche Aufgaben soll ein Nutzer oder eine Nutzerin typischerweise mit der App bewerkstelligen können?
- Ziel:
 - Produkt ist ein Prototyp aus Papier, mit Hilfe dessen das Grobkonzept gleichzeitig weiterentwickelt und überprüft (MB05 Usability Test) werden kann.
- Zugehörige Materialien:
 - selbst zu organisieren: Papier, Karton, Stifte, Klebe/Tesa, Schere, optional: Schablonen für Icons, Tablet-Rahmenlayout, Transparentfolie
- Weitere Quellen:
 - Schablonen, z.B. <https://www.uistencils.com/>
 - kurze Einführung (Engl.): <https://www.youtube.com/watch?v=4ZRzJTczMCE>
 - <https://de.wikipedia.org/wiki/Papierprototyp>
 - Paper Prototyping (Deutsch): <https://www.medien.ifi.lmu.de/lehre/ws0607/mmi1/essays/Guenther-Obexer.xhtml>
 - Paper Prototyping: The 10-Minute Practical Guide (Engl.): <https://www.uxpin.com/studio/blog/paper-prototyping-the-practical-beginners-guide/>

3.7. MB05: Usability-Testing: Papierprototypen

- Umfang:
 - ca. 4-8 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module MB01, MB04
- Kurzbeschreibung:
 - Die in Modul MB04 erstellten Papierprototypen werden auf ihre Usability (Nutzungsfreundlichkeit) getestet, der Test wird ausgewertet und Optimierungsmaßnahmen werden abgeleitet.
 -
- Ablauf:
 - Testaufbau und -durchführung werden erläutert und exemplarisch durchgespielt
 - Gruppen für die Testdurchführung bilden bzw. Personen der Zielgruppe(n) rekrutieren
 - Die Testperson bekommt Aufgaben gestellt und Probleme, aber auch Positives werden dokumentiert und ausgewertet
- Ziel:
 - Erkenntnisgewinn, an welcher Stelle Aspekte für die Zielgruppe noch nicht funktionieren. Beispiele dafür sind folgende:
 - Fehlende Buttons (z.B. zurück / undo)
 - Klickbare Elemente nicht sichtbar/groß genug
 - Navigationsstruktur nicht nachvollziehbar
 - Auf dieser Grundlage kann entweder der bestehende Papierprototyp überarbeitet werden oder die Erkenntnisse fließen in die nächsten Schritte ein (s. MB07, MB08)
- Zugehörige Materialien:
 - Checkliste Testing
 - Dokumentationsvorlage Testing
- Weitere Quellen:
 - kurzes Video zur Testdurchführung (Engl.):
<https://www.youtube.com/watch?v=9wQkLthhHKA>
 - <https://www.akendi.com/blog/how-to-do-a-paper-prototype-test-in-7-steps-the-quick-easy-and-dirt-cheap-way/> (Planung eines Tests mit Papierprototypen)

3.8. MB06: Programmierung: Grundlagen-Modul

- Umfang:
 - ca. 8-14 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module MB01, MB02
- Kurzbeschreibung:
 - Architektonische Gestaltung einer App oder Webseite
 - Theoretische Konzepte der Navigation (Hierarchisch, Flach oder Content/Experience-Driven)
 - Praktische Umsetzung einer Navigation in der App mit Seiten/Screens, Links und Tabs
- Ablauf:
 - Vorstellung der Diagrammart Flowchart und Analyse bestehender Webseiten/Apps zur Seitenstruktur durch die SuS. Erarbeitung der verschiedenen Navigationsarten (z.B. Hierarchisch, Flach oder Content/Experience-Driven).
 - Praktische Übungen um die Navigations-Elemente der UI-Bibliothek kennenzulernen. Anhand eines vorgegebenen Flowcharts sollen vorgegebene Seiten/Screens verbunden werden.
 - Es sollen Lösungen erarbeitet und umgesetzt werden, wie die vorhandene Navigationsstruktur optimiert werden kann.
- Ziel:
 - Eine Anwendung soll als Sammlung von Seiten/Screens verstanden werden, durch die die Nutzerinnen und Nutzer geführt werden müssen.
 - Es sollen die Anwendungsbereiche der einzelnen Navigationsarten erkannt werden und deren Auswirkungen sowohl auf Benutzerinnen und Benutzer, als auch die technische Umsetzung.
 - Es soll erkannt werden, dass die jeweiligen Navigationsarten auch miteinander kombiniert werden können.
- Zugehörige Materialien: /
- Weitere Quellen zum Modul:
 - <https://developer.apple.com/design/human-interface-guidelines/ios/app-architecture/navigation/> Navigation in iOS Apps
 - <https://material.io/design/navigation/understanding-navigation.html> Navigation in Material Design (Android und Flutter)
 - <https://app.diagrams.net/> Draw.io Erstellung von Diagrammen z.B. Flowcharts



3.9. MB07: Design: Grundlagen-Modul



- Umfang:
 - ca. 6-12 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module MB01, MB03
- Kurzbeschreibung:
 - Das Modul enthält den Einstieg in die visuelle Gestaltung. Die SuS werden befähigt, eigene Screen-Entwürfe aus fertigen Komponenten zu erstellen und Entwürfe zu kritisieren
 - Inhalte: Wahrnehmung, Gestaltgesetze, Raster/Komposition, Farbenlehre, Schriften/Typografie
 - Human Interface Guidelines (HIG), Paradigmen/Systeme (z.B. Material Design, Skeuomorphismus)
- Ablauf:
 - Nach dem Theorie-Input werden anhand der Grundlagen der Gestaltung, den Gestaltregeln und eines gewählten Designsystems (z.B. Apple Guidelines, s.u.) eigene Entwürfe digital erstellt. Alternativ können bisherige Entwürfe überarbeitet und verbessert werden.
 - Entlang der vorgestellten Elemente im Design werden die Entwürfe aufgebaut vom Groben zum Feinen
- Ziel:
 - Die SuS lernen Grundsätze der Gestaltung kennen und anzuwenden
 - Das Auge wird geschult, so dass Kritik formuliert und der Entwurf verbessert werden kann
- Zugehörige Materialien: /
- Weitere Quellen:
 - Apple: <https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/>
 - Google: <https://design.google/resources/>
 - Android <https://material.io/design>
 - Microsoft: <https://developer.microsoft.com/en-us/fluentui#/>
 - Design Best Practice/Patterns: <http://ui-patterns.com/patterns>
 - Software z.B.: Adobe Photoshop, Keynote, Sketch, Axure, Gimp
 - Sketch Educational License (free): <https://www.sketch.com/store/edu/>
 - Axure Educational License (free): <https://www.axure.com/edu>
 - <https://color.adobe.com/de/create/color-wheel> (Farbkompositionen mit Adobe erstellen)
 - Kostenlose Schriftarten: <https://www.dafont.com/de/>
 - Kostenlose Icons: <https://thenounproject.com/>

- Kompendium der Mediengestaltung
<https://link.springer.com/book/10.1007/978-3-540-78526-2>
- Handbuch visuelle Mediengestaltung: <https://www.visuelle-mediengestaltung.de/textbook-home.html>

3.10. MB08: Klick-Prototypen



- Umfang:
 - ca. 8-14 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Modul MB01, MB03, MB04
- Kurzbeschreibung:
 - Auf Basis des Papierprototypen und der Usability Testergebnisse (Modul MB04, MB05) wird in diesem Modul der Klickprototyp erstellt. Ein Klick-Dummy ist ein interaktiver Prototyp einer Webseite oder einer Applikation. Er weist bereits relevante Merkmale des fertigen Produktes auf, erhebt aber keinen Anspruch auf Vollständigkeit.
- Ablauf:
 - Die Grundlagen des Prototypen-Programms "Axure" werden vermittelt.
 - Erstellen des Screens im notwendigen Format (Größe)
 - Verwendung von Widgets zur Prototypen- und Wireframeerstellung
 - Hinzufügen von Interaktion (Screens werden in einen Ablauf gebracht und verlinkt)
 - Erstellen von wiederkehrenden Elementen (Master)
 - Hinterlegen von Animationen
- Ziel:
 - Am Ende dieses Moduls steht ein Prototyp, mit dem typische Interaktionsabläufe abgebildet sind.
- Zugehörige Materialien:
 - Muss selber ausgewählt und bereitgestellt werden: Programm für Klickdummys (beispielsweise Axure RP9 / RP10; diverse Alternativen existieren: Balsamiq, Adobe Xd, Figma, ...)
- Weitere Quellen:
 - <https://www.axure.com/edu> (free edu licence)
 - <https://developer.apple.com/design/resources/>
 - <https://developer.apple.com/videos/play/wwdc2020/10206/>
 - Explizit zu Axure RP:
 - Von Axure selber: „Getting Started with Axure RP“
<https://docs.axure.com/axure-rp/reference/getting-started-video/>
auf englisch, führt in das Programm ein
 - Youtube-Kanal „vaexperience“
https://www.youtube.com/channel/UCMEQZDhP_k_YkFnwoCgT7SQ
englisch, führt quasi komplett durch das Programm
 - Getting started with Axure Prototyping | Axure RP: Noob to Master Ep1 <https://www.youtube.com/watch?v=vzV2d2KxzWI>

- Convert Design to Prototype | Axure RP: Noob to Master Ep2
<https://www.youtube.com/watch?v=csWWTjOrQvk>
- Tipps für das Arbeiten mit Axure von Indra Burkart
https://www.youtube.com/watch?v=FYvAJkNk_nw (deutsch, 3 Tipps (Stylesheets, Master, Widget Libraries))

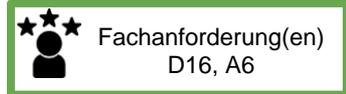
3.12. MB09: Usability-Testing: Klickprototypen

- Umfang:
 - ca. 3-6 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module Modul MB01, MB03, MB04, MB08
- Kurzbeschreibung:
 - Die erstellten Klickprototypen (Modul MB08) sollen mit Testpersonen getestet werden. Es werden Aufgaben erstellt, welche die Testperson mit dem Prototypen erfüllen soll.
- Ablauf:
 - Typische Aufgaben, die mit dem Prototyp bearbeitet werden können, herausarbeiten und notieren
 - Methode "Lautes Denken" vorstellen und einüben (<https://www.usability.de/usability-user-experience/glossar/concurrent-think-aloud.html>)
 - Einen Ablaufplan für den Test erstellen: Begrüßung, Test durchführen, ggf. anschließendes Interview oder Fragebogen ausfüllen, bedanken, verabschieden
 - Die Prototypen entsprechend der Aufgaben vorbereiten, möglichst auf das entsprechende Gerät (Computer, Tablet, Smartphone) spielen und der Testperson zur Verfügung stellen
 - Der Testleiter/die Testleiterin und der Protokollant/die Protokollantin beobachten und machen Notizen; ggf. können Audio-/Videoaufnahmen für die spätere Auswertung und zu Präsentationszwecken angefertigt werden
 - Hilfestellung für die Beobachtungen:
Welche Funktionen werden schnell und einfach gefunden, welche fehlen, wo ist die Navigationsstruktur noch nicht stimmig, wie wirkt sich das Design auf die Lesbarkeit und den Eindruck für die Testperson aus?
- Ziel:
 - Erkenntnisgewinn, an welcher Stelle Aspekte des Klickprototypen für die Zielgruppe noch nicht funktionieren bzw. gut funktionieren. Beispiele sind folgende:
 - Navigationsstruktur bzw. Informationsarchitektur sind nicht ausreichend nachvollziehbar
 - Interaktionsmöglichkeiten (z.B. von Buttons, Eingabefeldern) werden nicht als solche erkannt (Affordanzen)
 - Das Design skaliert nicht ausreichend auf die Bildschirmgröße des Zielgeräts (z.B. im Fall von Smartphones oder Tablets)
 - Auf dieser Grundlage kann entweder der bestehende Klickprototyp überarbeitet werden oder die Erkenntnisse fließen in die nächsten Schritte ein (s. MB10, MB11, MB12)

- Zugehörige Materialien:
 - / (siehe ansonsten Materialien aus MB05)

- Weitere Quellen zum Modul:
 - <https://www.nngroup.com/articles/usability-testing-101/>
 - <https://axuretips.com/how-to-do-user-testing-on-axure-prototype/>

3.13. MB10: Programmierung: Vertiefungs-Modul



- Umfang:
 - ca. 7-14 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Modul MB02, MB06
- Kurzbeschreibung:
 - Wiederverwendbare Designs
 - Zustandsabhängige Designs
 - Animationen (Interaktion)
 - Übergänge/Transitions
- Ablauf:
 - In einer vorgegeben Anwendung sollen Doppelungen von gleichen UI-Elementen erkannt werden. Diese Doppelungen sollen in wiederverwendbare eigene UI-Elemente ausgelagert und mit den Doppelungen ersetzt werden.
 - Die eigenen UI-Elemente sollen abgeändert werden, sodass sie abhängig von einem Wert (z.B. Button aktiv/inaktiv) ein unterschiedliches Design erhalten. Darauf aufbauend werden Animation angewendet, um den Übergang von einem Zustand in den anderen ansprechender zu gestalten.
 - Als Zusatzübung, für fleißige SuS, werden noch Übergänge von einer Seite/Screen zu einer anderen Seite/Screen gezeigt.
- Ziel:
 - Die SuS sollen die eigenen designten UI-Elemente als wiederverwendbare Bausteine begreifen und nutzen können. Diese Bausteine können bereits auf Benutzeraktionen und Zustände reagieren und sich abhängig davon darstellen.
 - Die Zustandsänderung kann dann mit einer Animation versehen werden, um die Benutzeraktion hervorzuheben und einen flüssigen Übergang der Zustände herzustellen.
 - Weitere Besonderheiten der UI-Bibliothek sind den SuS bekannt und von ihnen genutzt werden, um effizienter den nativen Prototypen zu erstellen.
- Zugehörige Materialien: /
- Weitere Quellen zum Modul:
 - <https://www.droptica.com/blog/component-based-design/> Blogartikel über Vorteile von komponentenbasierten Designs

3.14. MB11: Design: Vertiefungsmodul

- Umfang:
 - ca. 4-12 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen: MB01, MB03, MB07, MB08
- Kurzbeschreibung:
 - Bei der Vertiefung des Designs nähern sich die SuS weiter an das echte Produkt an. Dies kann auf verschiedene Weise geschehen und hängt stark von den vorherigen Arbeitsschritten ab.
- Ablauf:
 - Die Vertiefung kann anhand des Klickprototyps geschehen (Ziel: High-Fidelity Prototyp) oder ein separates Designdokument sein (z.B. in Photoshop); ggf. kann auch schon direkt programmiert werden
 - Erkenntnisse aus den Usability Tests einfließen lassen
 - Interaktionsabläufe und Umfang werden erweitert, z.B. mögliche Anwendungsfehler antizipieren und konzeptionell abfangen
 - Adäquate Animationen/Transitionen entwickeln, d.h. unterstützende Maßnahmen für die Benutzung (z.B. Steuern der Aufmerksamkeit, Mouseover-Effekte, Lade-Animation o.ä.)
 - Produktion der benötigten Grafiken für die Software-Entwicklung, sogenannte Asset-Produktion. Hierzu bieten viele Design-System-Autorinnen und -Autoren Dokumente an, s. Quellen
 - Das Wording in der Anwendung wird geprüft, ggf. überarbeitet und einheitlich umgesetzt
- Ziel:
 - Animationen erstellen und adäquat einsetzen
 - Überführen des Design-Entwurfs in die Implementierung
 - flankierende Designtätigkeiten identifizieren und umsetzen (z.B. App-Icon für den App Store gestalten, Tutorial erstellen u.ä.)
- Zugehörige Materialien: /
- Weitere Quellen:
 - Produktionsvorlagen von Apple:
<https://developer.apple.com/design/resources/>
 - <https://developer.apple.com/tutorials/swiftui/animating-views-and-transitions>
 - UX Writing <https://xd.adobe.com/ideas/process/information-architecture/ux-writing-guidelines/>

3.15. MB12: Produkt: Nativer Programmier-Prototyp

- Umfang:
 - ca. 7-14 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Modul MB02, MB06, effizienter mit Modul MB10
 - Modul MB 04, MB08
- Kurzbeschreibung:
 - Nutzung der bisheriger Programmierkenntnisse zur Umsetzung des Papier- oder Klickprototypens.
 - Je nach Programmierkenntnis der SuS kann der Programmier-Prototyp einfacher oder detailreicher ausfallen.
- Ablauf:
 - Der Papier- oder Klickprototyp wird zunächst in Seiten/Screens aufgeteilt und als Flowchart eine logische Verbindung zwischen diesen Seiten/Screens hergestellt.
 - Die einzelnen Seiten/Screens werden erstellt. Je nach Fähigkeit der SuS und Zeitrahmen einfach erstellt oder mit wiederverwendbaren eigenen UI-Elementen.
 - Die erstellten Seiten/Screens werden mit der passenden Navigationsstruktur verbunden.
 - Je nach Zeit und Fähigkeit kann der Programmier-Prototyp noch mit Animationen und Übergängen versehen werden.
- Ziel:
 - Praktische Anwendung aller erlernten Programmierfähigkeiten und Umsetzung der Vorarbeiten durch Interviews, Papier-, und Klick-Prototypen als programmierte Anwendung.
 - Erkenntnis der unterschiedlichen Sichtweise einer Anforderung aus Benutzerinnen- und Benutzer- sowie Entwicklerinnen und Entwickler-Sicht. Erstes Gefühl für den Aufwand zur Umsetzung einer Benutzerinnen- und Benutzer-Anforderung.
- Zugehörige Materialien:
 - selbst erstellt im Projekt: fertiger zu testender Papierprototyp oder Klickprototyp
- Weitere Quellen zum Modul: /



3.16. MB13: Usability-Testing: Produkt: Nativer Prototyp

- Umfang:
 - ca. 3-6 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module MB01, MB02, MB04, MB05, MB06, MB10, MB12
- Kurzbeschreibung:
 - In diesem Modul werden die im Modul MB12 erstellten, nativen Prototypen mit Personen der Zielgruppe(n) auf ihre Usability geprüft. Auf Basis der gewonnenen Daten werden Optimierungsmaßnahmen abgeleitet.
- Ablauf:
 - Der generelle Ablauf ist vergleichbar dem aus MB09, in welchem die Klickprototypen getestet wurden (siehe daher bitte dort).
 - So wird insbesondere erneut entlang von typischen Aufgaben, die mit dem Prototyp erledigt werden sollen, getestet.
 - Auch kommt die Methode des lauten Denkens zum Einsatz.
 - Als Option: Die abgeleiteten Optimierungsmaßnahmen können bei Bedarf erneut im MB12 geleitet werden.
- Ziel:
 - Erkenntnisgewinn, an welcher Stelle Aspekte des nativen Prototypen für die Zielgruppe noch nicht funktionieren bzw. gut funktionieren. Siehe bitte MB09 für weitere Details.
- Zugehörige Materialien: / (siehe dazu Materialien aus MB05)
- Weitere Quellen zum Modul: /

3.17. MB14: Abschlussmodul: Projektabschluss/ Ergebnispräsentation

- Umfang:
 - ca. 2-6 Unterrichtseinheiten à 45 Minuten
- Vorbedingungen:
 - Module 01, 03, ebenfalls sinnvoll: 04 und 08
- Kurzbeschreibung:
 - Dieses Modul stellt keine neuen Inhalte mehr da, sondern dient der gemeinsamen Sichtung der Produkte, sprich der Zwischenergebnisse und Prototypen, die im Laufe der Projektzeit entstanden sind.
- Ablauf:
 - UserFlowcharts: Foto-Präsentation am Beamer, Gespräch im Klassenverbund
 - Papierprototypen: Auslage im Klassenraum (Galerie), anschließende Diskussion im Klassenverbund
 - Klickprototypen: kurze 5-Minuten-Präsentation der Highlights per Screensharing am Beamer
 - Produkt nativer Prototyp: kurze 5-Minuten-Präsentation der Highlights per Screensharing am Beamer
- Ziel:
 - Es soll für alle Schülerinnen und Schüler eine Übersicht entstehen, was sie selbst und die anderen Gruppen geschaffen und erschaffen haben. Das Erfolgserlebnis, selber ein Produkt gestaltet, entwickelt und programmiert zu haben steht hier im Vordergrund.
 - Auch die Reflektion des Umgangs mit Medien (Programme, nötige Vorarbeit, notwendiges Hintergrundwissen, Aufwand) gehört zu den erworbenen Kompetenzen dieses Projektes. Ebenso wie die Erkenntnis, dass eine App, Webseite oder ähnliches entsprechend gut geplant, konzipiert, gestaltet und im Anschluss getestet werden muss, damit am Ende ein nutzerfreundliches Produkt stehen kann.
- Zugehörige Materialien:
 - Alle Produkte, Prototypen und Entwürfe der Schülerinnen und Schüler
- Weitere Quellen zum Modul: /

3.18. Ausgewählte Beispiele für die Kombination der Modulbausteine

Im folgenden sind drei beispielhafte Darstellung für die Zusammensetzung der einzelnen Modulbausteine dargestellt, um zu zeigen, wie die Handlungsempfehlung auf die Bedürfnisse der Lehrkraft und der Schülerinnen und Schüler sowie die Bedingungen des Unterrichts (Stundenumfang) angepasst werden kann.

Abbildung 4: Variante 1: Schwerpunkt auf die Programmierung und das Prototypen bauen

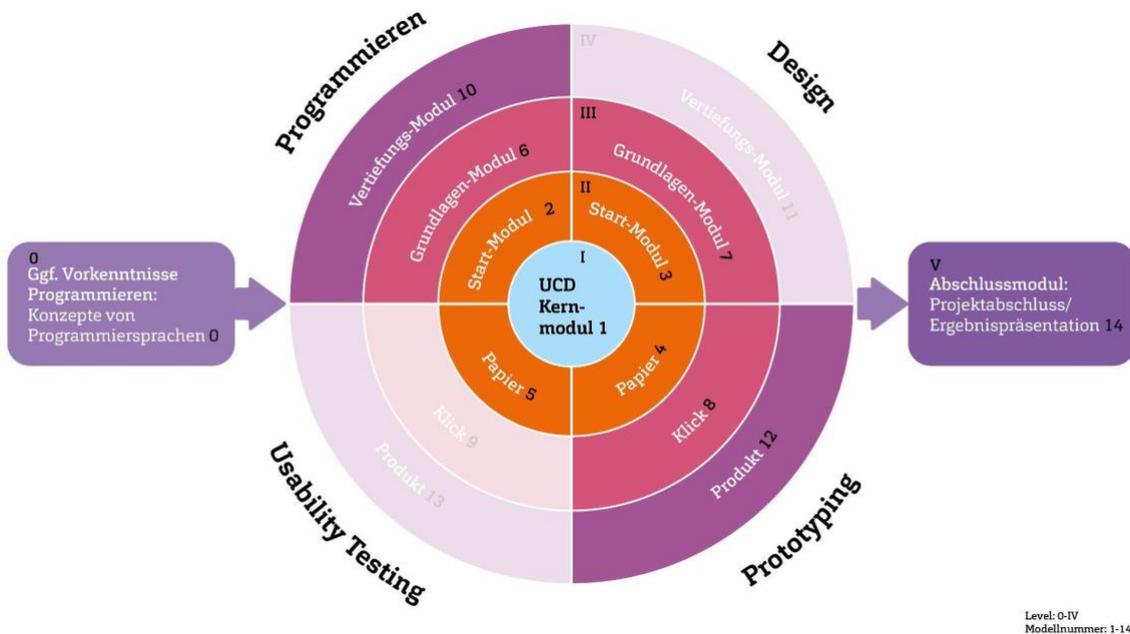


Abbildung 5: Variante 2: Schwerpunkt auf das Design und das Prototyping

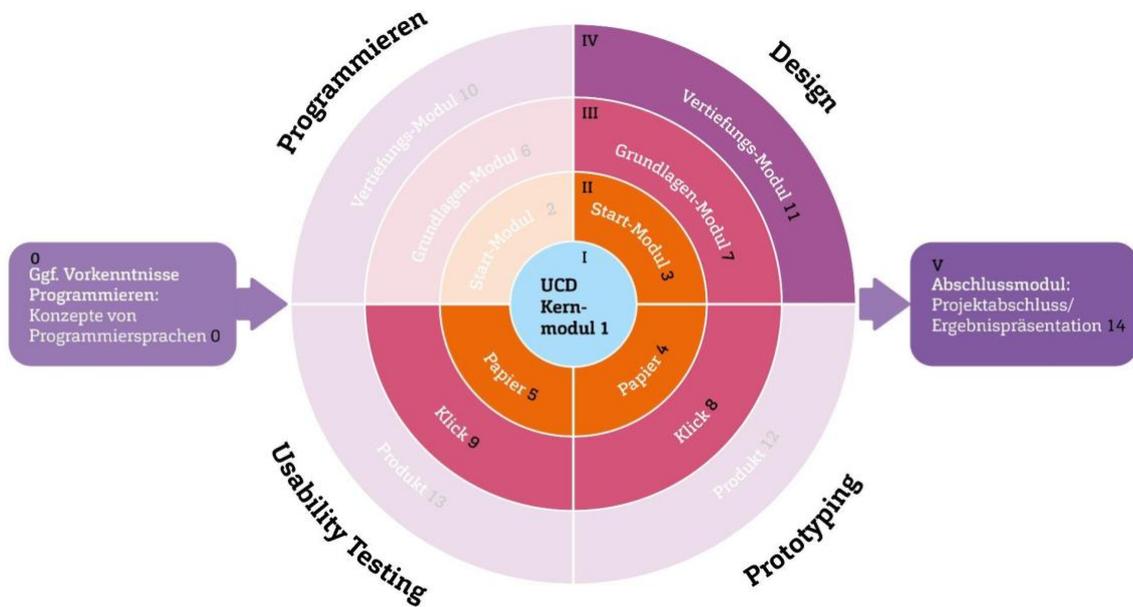
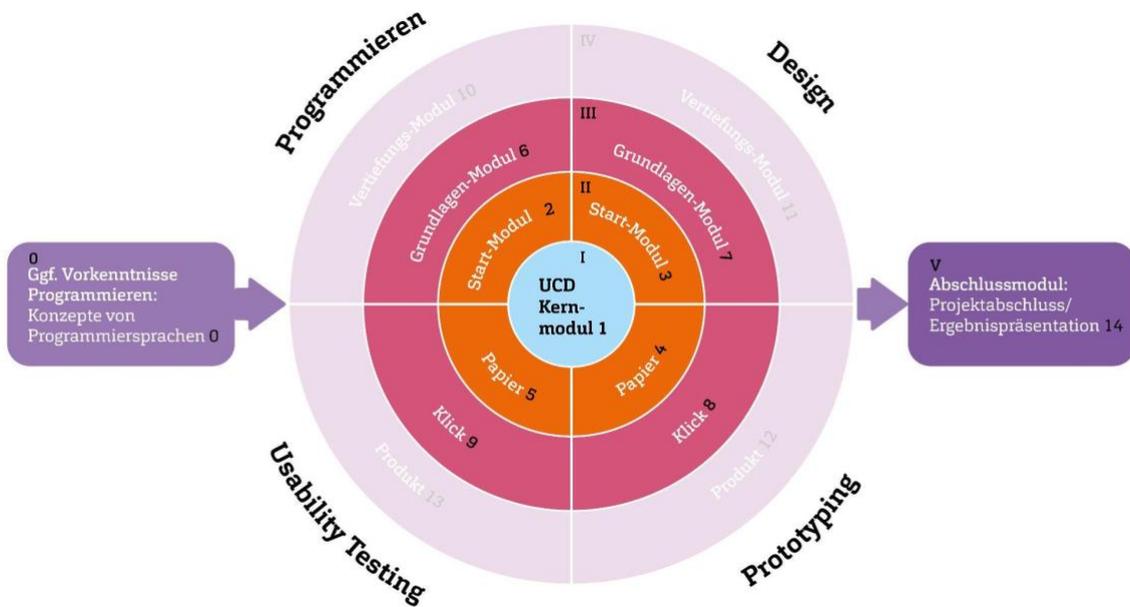


Abbildung 6: Variante 3: Schwerpunkt auf die ersten beiden Zyklen des UCD-Prozesses



4. Verknüpfung mit den Kompetenzen in den Fachanforderungen



Tabelle 1: Referenzen zu den Kompetenzen in den Fachanforderungen

<u>Handlungsempfehlung</u>			<u>Fachanforderungen</u>	
<u>Index</u>	<u>Name</u>	<u>Beschreibung</u>	<u>Index</u>	<u>Die Schülerinnen und Schuler...</u>
MB01	Kernmodul UCD	In diesem Modul werden die grundlegenden Prozesse im User Centered Design-Prozess erläutert und der Überblick über die Schritte des Zyklus vermittelt.	SE3	... nennen Vorgehensmodelle und bewerten die Eignung für ein Vorhaben.
MB02	Programmierung Einführung	Anforderungen System / Einführung in die Programmiersprache und erste Gestaltung anhand von kleinen Beispielaufgaben / Aufbau von Stacks, horizontal/vertikal / Elemente auf einer Profilsseite anordnen	D15	... entwickeln aus einer Problemstellung eine passende Dokumentstruktur
			D21	... beschreiben Elemente von Dokumenten und Dateisystemen als Objekte mit Attributen und Werten
			I13	... verwenden grundlegende Funktionen des Betriebssystems zur Bewältigung typischer Aufgaben
MB03	Design-Einführung	Dieses Modul dient dazu, den Schülerinnen und Schülern ein Bewusstsein für die Bedeutung und Relevanz von Design von (Software-) Produkten zu vermitteln. Es wird sich dabei zuerst mit auseinandergesetzt, was hinter den Begriffen Design, Interface und Affordance steckt und welche Bedeutung diese Aspekte für Alltag, Produktdesign	I5	... beschreiben und bewerten den Einfluss von Informatiksystemen auf ihre Lebenswelt.

		und Softwareentwicklung haben.		
MB04	Papier-Prototypen	In diesem Modul werden die Schülerinnen und Schüler ihre ersten eigenen Prototypen erstellen. In diesem Durchgang des UCD-Zyklus sind es Papierprototypen, die gebaut werden und später im Modul MB05 getestet werden können.	SE2 SE7 SE8	... kommunizieren mit Dritten und innerhalb eines Teams über einen Entwicklungsprozess und geplante Systeme ... entwickeln Modelle eines Systems basierend auf gestellten Anforderungen ... verwenden geeignete Beschreibungsformate, um solche Modelle darzustellen
MB05	Usability-Testing: Papierprototypen	Nachdem im Modul MB04 die Papierprototypen gebaut wurden sollen sie nun auf ihre Nutzerfreundlichkeit und Usability getestet werden. Das Testing-Prozedere wird im Folgenden beschrieben und Anweisungen für die Auswertung gegeben.		
MB06	Programmierung: Grundlagen	erste Gestaltung einer App oder Webseite / Navigation (Hierarchisch, Flach oder Content/Experience-Driven) / SwiftUI Views: NavigationView, NavLink und TabBar	A5 / AD1 AD8 AD17	... beschreiben sowohl für einzelne Anweisungen wie auch für Algorithmen im Ganzen das Ergebnis der Ausführung ... formulieren Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine ... verwenden vorgegebene Klassen und Objekte bei der Programmierung
MB07	Design-Grundlagen	Human-Interface-Guidelines, Farbenlehre, Schriften, Raster/Komposition, Visual Design, (Material Design, Neumorphism, Skeumorphismus)	D16	... verwenden Formatvorlagen zweckmäßig und sparsam

MB08	Klick-Prototyp	Auf Basis des Papierprototypen (Modul MB04) wird in diesem Modul der Klickprototyp erstellt. Dieser beinhaltet die Ergänzungen und Veränderungen auf Basis der Erkenntnisse aus dem Usability Testing (Modul MB05).	SE7	... entwickeln Modelle eines Systems basierend auf gestellten Anforderungen
			SE8	... verwenden geeignete Beschreibungsformate, um solche Modelle darzustellen
MB09	Usability-Testing: Klickprototypen	Auch die erstellten Klickprototypen (Modul MB08) sollen getestet werden. Dabei werden die Prototypen auf das entsprechende Medium (Computer, Tablet, Smartphone) gespielt und werden dann durch eine Testperson durchgeklickt. Der Animator /die Animatorin und der Protokollant/die Protokollantin beobachten den Klickprozess...		
MB10	Programmierung: Vertiefung	Wiederverwendbare Designs / Zustandsabhängige Designs / Animationen (Interaktion) / Übergänge / Transaktionen	D16	... verwenden Formatvorlagen zweckmäßig und sparsam
			A6	... implementieren einfache Algorithmen in einer grafischen Programmierumgebung
MB11	Design-Vertiefung	Animationen (Interaktion)		
MB12	Nativer Programmier-Prototyp	Umsetzung aller bisheriger Kenntnisse, Umsetzung des Papier- und/oder Klickprototypen im Programmierung mitsamt allen Verbesserungen/Änderungen	A11	... analysieren eine algorithmische Problemstellung, um Teilprobleme zu identifizieren
			AD9	... entwerfen und implementieren Algorithmen zur Lösung einer gegebenen Problemstellung

			SE10	... implementieren auf Basis einer Modellierung eine Software unter Verwendung geeigneter objektorientierter Techniken
MB13	Usability-Testing: Nativer Prototyp		SE11	... beschreiben Verfahren, um komplexen Code zu testen
MB14	Projektabschluss: Ergebnispräsentation	Dieses Modul stellt keine neuen Inhalte mehr da, sondern dient der gemeinsamen Sichtung der Produkte, sprich der Zwischenergebnisse und Prototypen, die im Laufe der Projektzeit entstanden sind.		

Tabelle 2: Übersicht zu den inhaltsbezogenen Kompetenzen aus den Fachanforderungen

Sek I

<u>Index</u>	<u>Beschreibung</u>
D15	... entwickeln aus einer Problemstellung eine passende Dokumentstruktur
D16	... verwenden Formatvorlagen zweckmäßig und sparsam
D21	... beschreiben Elemente von Dokumenten und Dateisystemen als Objekte mit Attributen und Werten
I5	... beschreiben und bewerten den Einfluss von Informatiksystemen auf ihre Lebenswelt.
I13	... verwenden grundlegende Funktionen des Betriebssystems zur Bewältigung typischer Aufgaben
I15	... konfigurieren Software zielorientiert und ergonomisch

I16	... verwenden Programm- oder Online-Hilfen zur Benutzung
I17	... beschreiben Fehler
I18	... ordnen Fehler Hard- bzw. Software zu
I19	... verwenden Hilfen zur Fehlerdiagnose
I20	... untersuchen fehlerhafte Systeme, bestimmen Ursachen und wählen Lösungsansätze aus
A3	... formulieren Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine
A5	... beschreiben sowohl für einzelne Anweisungen wie auch für Algorithmen im Ganzen das Ergebnis der Ausführung
A6	... implementieren einfache Algorithmen in einer grafischen Programmierumgebung
A10	... modellieren eine algorithmische Problemstellung aus einem Sachkontext
A11	... analysieren eine algorithmische Problemstellung, um Teilprobleme zu identifizieren
A13	... entwerfen und implementieren Algorithmen zur Lösung einer gegebenen Problemstellung
BK4	... erklären die grundlegende Funktionsweise von Hardwarekomponenten und deren Zusammenwirken
BK19	... beschreiben Webanwendungen und ihre Architektur
AD1	... beschreiben sowohl für einzelne Anweisungen wie auch für Algorithmen im Ganzen das Ergebnis der Ausführung
AD4	... vergleichen und beurteilen Algorithmen zum Lösen eines Problems
AD8	... formulieren Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine
AD9	... entwerfen und implementieren Algorithmen zur Lösung einer gegebenen Problemstellung
AD17	... verwenden vorgegebene Klassen und Objekte bei der Programmierung
SE1	... analysieren die von Dritten gestellten Anforderungen an ein zu entwickelndes System
SE2	... kommunizieren mit Dritten und innerhalb eines Teams über einen Entwicklungsprozess und geplante Systeme
SE3	... nennen Vorgehensmodelle und bewerten die Eignung für ein Vorhaben.
SE4	... erstellen Zeit- und Arbeitspläne für Projektarbeit

- SE7 ... entwickeln Modelle eines Systems basierend auf gestellten Anforderungen
- SE8 ... verwenden geeignete Beschreibungsformate, um solche Modelle darzustellen
- SE9 ... nennen Strukturmuster und Architekturen und erklären ihre Verwendung in existierenden Systemen
- SE10 ... implementieren auf Basis einer Modellierung eine Software unter Verwendung geeigneter objektorientierter Techniken
- SE11 ... beschreiben Verfahren, um komplexen Code zu testen

5. Literaturverzeichnis

- Bundesministerium für Bildung und Forschung (BMBF). (17. Mai 2019): DigitalPakt Schule 2019 bis 2024. *Verwaltungsvereinbarung vom 17.05.2019*. Deutschland.
- Bundesministerium für Bildung und Forschung (BMBF). (30. April 2021): *DigitalPakt Schule*. Von Was ist der DigitalPakt Schule?: <https://www.digitalpaktschule.de/de/was-ist-der-digitalpakt-schule-1701.html>
- Digital.gov team in the U.S. General Services Administration (GSA) Technology Transformation Service. (26. Juli 2021): *usability.gov*. Von www.usability.gov/what-and-why/user-interface-design: <https://www.usability.gov/what-and-why/user-interface-design.html> abgerufen
- Kultusministerkonferenz (KMK). (08. Dezember 2016): Strategie der Kultusministerkonferenz "Bildung in der digitalen Welt". *Beschluss der Kultusministerkonferenz vom 08.12.2016 in der Fassung vom 07.12.2017*. Deutschland.
- Schleswig-Holstein Ministerium für Bildung, Wissenschaft und Kultur. (Juli 2018): Ergänzung zu den Fachanforderungen. *Medienkompetenz - Lernen mit digitalen Medien (Allgemein bildende Schulen, Sekundarstufe 1, Sekundarstufe 2)*. Kiel, Schleswig-Holstein, Deutschland.
- Schleswig-Holstein Ministerium für Bildung, Wissenschaft und Kultur. (14. August 2020): Informatik Sek 1 und Sek 2. *Verbindliche Gliederung und Allgemeiner Teil Stand 14.08.2020 (Anhörungsfassung)*. Kiel, Schleswig-Holstein, Deutschland.
- Usability.gov. (27. August 2021): *Usability.gov*. Von Usability.gov -User Interface Design Basics: <https://www.usability.gov/what-and-why/user-interface-design.html>

6. Graphikverzeichnis:

Abbildung 1: Prozessbezogene Kompetenzen (Fachanforderungen Informatik SekI+II, IQSH, S.13).....	6
Abbildung 2: Visuelle Übersicht der Module	11
Abbildung 3: Iterativer Prozess: UCD als Zyklus.....	14
Abbildung 4: Variante 1: Schwerpunkt auf die Programmierung und das Prototypen bauen	33
Abbildung 5: Variante 2: Schwerpunkt auf das Design und das Prototyping	34
Abbildung 6: Variante 3: Schwerpunkt auf die ersten beiden Zyklen des UCD-Prozesses.....	35

7. Tabellenverzeichnis

Tabelle 1: Referenzen zu den Kompetenzen in den Fachanforderungen.....	36
Tabelle 2: Übersicht zu den inhaltsbezogenen Kompetenzen aus den Fachanforderungen	39