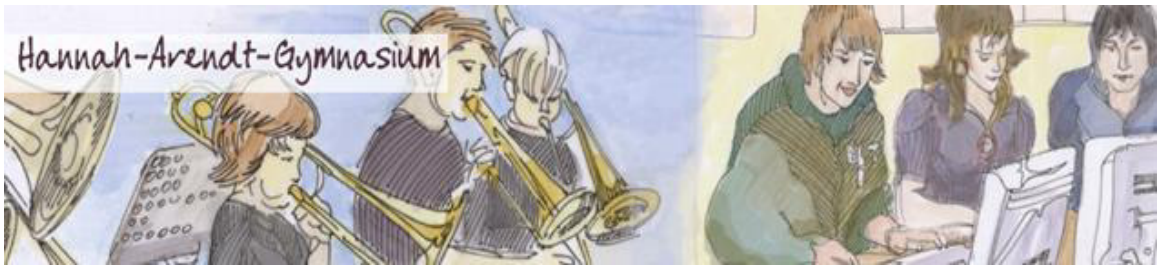

H a n n a h - A r e n d t - G y m n a s i u m

49525 Lengerich, Hannah-Arendt_Str. 2

Schulinterner Lehrplan

Informatik, Oberstufe

Stand: August 2025

Inhaltsverzeichnis

Entscheidungen zum Unterricht 3
Unterrichtsvorhaben 3
EF-I: Einführung in die Nutzung von Informatiksystemen und grundlegende Begrifflichkeiten 4
EF-II: Grundlagen der objektorientierten Analyse, Modellierung und Implementierung 4
EF-III: Grundlagen der objektorientierten Programmierung und algorithmischer Grundstrukturen 5
EF-IV: Modellierung und Implementierung von Klassen- und Objektbeziehungen 5
EF-V: Such- und Sortieralgorithmen anhand kontextbezogener Beispiele 5
EF-VI: Geschichte der digitalen Datenverarbeitung und Grundlagen des Datenschutzes 6
Q1-I: Wiederholung der objektorientierten Modellierung und Programmierung 7
Q1-II: Anwendungen mit dynamischen, linearen Datenstrukturen 7
Q1-III: Suchen und Sortieren auf linearen Datenstrukturen 8
Q1-IV: Modellierung und Nutzung relationaler Datenbanken 8
Q1-V: Sicherheit und Datenschutz in Netzstrukturen 9
Q2-I: Anwendungen mit dynamischen, nichtlinearen Datenstrukturen 9
Q2-II: Endliche Automaten und formale Sprachen 10
Q2-III: Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit 11
Grundsätze der fachdidaktischen und fachmethodischen Arbeit 12
Lehr- und Lernmittel 12
Qualitätssicherung und Evaluation 13
Maßnahmen der fachlichen Qualitätssicherung 13
Überarbeitungs- und Planungsprozess 13

Entscheidungen zum Unterricht

Unterrichtsvorhaben

In der nachfolgenden Übersicht über die Unterrichtsvorhaben wird die für alle Lehrerinnen und Lehrer gemäß Fachkonferenzbeschluss verbindliche Verteilung der Unterrichtsvorhaben dargestellt. Die Übersicht dient dazu, für die einzelnen Jahrgangsstufen allen am Bildungsprozess Beteiligten einen schnellen Überblick über Themen bzw. Fragestellungen der Unterrichtsvorhaben unter Angabe besonderer Schwerpunkte in den Inhalten und in der Kompetenzentwicklung zu verschaffen. Dadurch soll verdeutlicht werden, welches Wissen und welche Fähigkeiten in den jeweiligen Unterrichtsvorhaben besonders gut zu erlernen sind und welche Aspekte deshalb im Unterricht hervorgehoben thematisiert werden sollten. Unter den Hinweisen des Übersichtsrasters werden u.a. Möglichkeiten im Hinblick auf inhaltliche Fokussierungen und interne Verknüpfungen ausgewiesen.

Der zeitliche Umfang der einzelnen Vorhaben folgt grob der Einteilung des Schuljahres in Quartale, so dass jedes Vorhaben je nach Schwerpunktsetzung des Fachlehrers und in Absprache mit den Schülerinnen und Schülern zwischen 8 und 12 Wochen beträgt. Der Zeitbedarf versteht sich als grobe Orientierungsgröße, die nach Bedarf über- oder unterschritten werden kann. Der schulinterne Lehrplan ist so gestaltet, dass er zusätzlichen Spielraum für Vertiefungen, besondere Interessen von Schülerinnen und Schülern, aktuelle Themen bzw. die Erfordernisse anderer besonderer Ereignisse (z.B. Praktika, Klassenfahrten o.Ä.) lässt. Abweichungen über die notwendigen Absprachen hinaus sind im Rahmen des pädagogischen Gestaltungsspielraumes der Lehrkräfte möglich. Sicherzustellen bleibt allerdings auch hier, dass im Rahmen der Umsetzung der Unterrichtsvorhaben insgesamt alle Kompetenzerwartungen des Kernlehrplans Berücksichtigung finden

EF-I: Einführung in die Nutzung von Informatiksystemen und grundlegende Begrifflichkeiten

Leitfragen

- Was ist Informatik und wie nutzen wir Informatiksysteme im Alltag?
- Wie funktioniert ein Rechner und wie kommunizieren Systeme miteinander?

Inhaltsschwerpunkte und Verlauf

- EVA-Prinzip und Von-Neumann-Architektur
- Sender-Empfänger-Modell und Netzwerke
- Dateisysteme und Internetnutzung

Diese Einheit bietet einen Einstieg in das Fach Informatik und vermittelt grundlegende Begriffe und Konzepte. Die Schülerinnen und Schüler lernen, wie Informationen codiert, gespeichert und übertragen werden. Sie setzen sich mit dem Aufbau von Rechnern und Netzwerken auseinander und reflektieren die Nutzung der schulischen Informatiksysteme im Kontext von Datenschutz und verantwortungsvollem Umgang.

Kompetenzbereiche:

- Argumentieren (A): Die Schüler erläutern den Aufbau und die Arbeitsweise eines Rechnersystems anhand der Von-Neumann-Architektur.
- Darstellen und Interpretieren (DI): Sie stellen die Kommunikation in Netzwerken grafisch dar und interpretieren Datenflüsse.
- Kommunizieren und Kooperieren (KK): Sie nutzen das Schulnetzwerk zur strukturierten Datenverwaltung und arbeiten kooperativ in Gruppen.

EF-II: Grundlagen der objektorientierten Analyse, Modellierung und Implementierung

Leitfragen

- Wie lassen sich reale Gegenstandsbereiche informatisch modellieren?
- Was sind Objekte, Klassen und Methoden?

Inhaltsschwerpunkte und Verlauf

- Objektkarten, Klassenkarten, Beziehungsdiagramme
- Einführung in Java mit Greenfoot
- Statische Szenen ohne Kontrollstrukturen

Die Schülerinnen und Schüler lernen die Grundlagen der objektorientierten Programmierung kennen. Sie analysieren lebensnahe Beispiele, modellieren Objekte und Klassen und setzen diese in einfachen Java-Projekten um. Die didaktische Bibliothek Greenfoot ermöglicht eine visuelle Umsetzung ohne algorithmische Ablenkung.

Kompetenzbereiche:

- Modellieren (M): Die Schüler identifizieren Objekte, modellieren deren Eigenschaften und Beziehungen.
- Implementieren (I): Sie implementieren einfache Klassen und nutzen Methoden zur Objektmanipulation.
- Darstellen und Interpretieren (DI): Sie visualisieren Objektbeziehungen und dokumentieren Klassenstrukturen.

EF-III: Grundlagen der objektorientierten Programmierung und algorithmischer Grundstrukturen

Leitfragen

- ⍰ Wie entstehen Animationen durch Programmierung?

⍰ Wie steuert man Abläufe mit Schleifen und Bedingungen?

Inhaltsschwerpunkte und Verlauf

- ⍰ Kontrollstrukturen: Schleifen, Bedingungen
- ⍰ Arrays und Objektverwaltung
- ⍰ Animationen und Simulationen

In dieser Einheit erweitern die Schüler ihre Kenntnisse um algorithmische Grundstrukturen. Sie entwickeln interaktive Animationen und Simulationen, steuern Objekte über Tastatureingaben und verwalten sie in Datenstrukturen. Eigene Klassen mit Methoden und Zustandsvariablen werden eingeführt.

Kompetenzbereiche:

- Modellieren (M): Die Schüler entwerfen Klassen mit Attributen, Methoden und Sichtbarkeitsbereichen.
- Implementieren (I): Sie implementieren Algorithmen mit Kontrollstrukturen und testen Programme systematisch.
- Argumentieren (A): Sie analysieren und erläutern die Funktionsweise ihrer Programme.

EF-IV: Modellierung und Implementierung von Klassen- und Objektbeziehungen

Leitfragen

- Wie kommunizieren Objekte miteinander?
- Was bedeutet Vererbung in der Programmierung?

Inhaltliche Schwerpunkte und Verlauf

- Dynamische Referenzierung
- Assoziationen und Kollisionskontrollen
- Vererbung und späte Bindung

Die Schüler lernen komplexere Objektbeziehungen kennen. Sie steuern Objekte über Referenzen, entwickeln Spiele mit Kollisionskontrollen und setzen Vererbung zur Spezialisierung und Erweiterung von Klassen ein. Die Prinzipien der objektorientierten Programmierung werden vertieft.

Kompetenzbereiche:

- Modellieren (M): Die Schüler modellieren Klassen mit Vererbungs- und Assoziationsbeziehungen.
- Implementieren (I): Sie implementieren komplexe Objektbeziehungen und korrigieren Fehler im Quellcode.
- Darstellen und Interpretieren (DI): Sie dokumentieren Klassen und visualisieren Objektkommunikation.

EF-V: Such- und Sortieralgorithmen anhand kontextbezogener Beispiele

Leitfragen

- Wie kann man Daten effizient sortieren und durchsuchen?

Inhaltliche Schwerpunkte und Verlauf

- Sortieren durch Vertauschen, Auswählen, Quicksort

- Welche Algorithmen sind besonders leistungsfähig?

- Binäre Suche
- Effizienzbetrachtung

Die Schüler erarbeiten verschiedene Such- und Sortierverfahren, analysieren deren Effizienz und formulieren Algorithmen im Pseudocode. Die binäre Suche wird als Beispiel für rekursives Problemlösen eingeführt. Die Verfahren werden ohne Programmiersprache behandelt.

Kompetenzbereiche:

- Argumentieren (A): Die Schüler beurteilen die Effizienz von Algorithmen.
- Modellieren (M): Sie entwickeln eigene Sortierverfahren und beschreiben sie grafisch.
- Darstellen und Interpretieren (DI): Sie interpretieren algorithmische Abläufe und vergleichen Verfahren.

EF-VI: Geschichte der digitalen Datenverarbeitung und Grundlagen des Datenschutzes

Leitfragen

- Wie hat sich die Datenverarbeitung entwickelt?
- Was bedeutet Datenschutz im digitalen Zeitalter?

Inhaltliche Schwerpunkte und Verlauf

- Historische Entwicklung der Informatik
- Datenschutzgesetz und Fallbeispiele
- Persönliche Daten und deren Schutz

Die Schüler recherchieren eigenständig zur Geschichte der Informatik und präsentieren ihre Ergebnisse. Anschließend vertiefen sie den Datenschutz anhand aktueller Fallbeispiele und diskutieren ethische Fragestellungen. Die Reflexion über den verantwortungsvollen Umgang mit Daten steht im Mittelpunkt.

Kompetenzbereiche:

- Argumentieren (A): Die Schüler bewerten Datenschutzsituationen und reflektieren gesellschaftliche Auswirkungen.
- Darstellen und Interpretieren (DI): Sie stellen historische Entwicklungen und Datenschutzkonzepte dar.
- Kommunizieren und Kooperieren (KK): Sie präsentieren ihre Ergebnisse und diskutieren im Plenum.

Qualifikationsphase

Q1-I: Wiederholung der objektorientierten Modellierung und Programmierung

Leitfragen
- Wie modelliert und implementiert man Klassen und Beziehungen in Java?
- Wie stellt man Objektkommunikation grafisch dar?

Inhaltliche Schwerpunkte und Verlauf

- Klassen, Vererbung, abstrakte Klassen
- Implementationsdiagramme
- Objektkommunikation und Sichtbarkeiten

Die Schüler wiederholen zentrale Konzepte der objektorientierten Programmierung. Sie analysieren eine Problemstellung, modellieren Klassen mit Vererbungsbeziehungen und setzen diese in Java um. Die Kommunikation zwischen Objekten wird grafisch dargestellt und dokumentiert. Die Anwendung wird getestet und reflektiert.

Kompetenzbereiche:

- Modellieren (M): Modellierung von Klassen mit Attributen, Methoden und Vererbungsbeziehungen.
- Implementieren (I): Umsetzung der Modellierung in Java unter Nutzung dokumentierter Klassenbibliotheken.
- Darstellen und Interpretieren (DI): Visualisierung von Objektkommunikation und Dokumentation von Klassen.
- Argumentieren (A): Analyse und Bewertung der Modellierung und Funktionalität.

Leistungskurs-Erweiterung:

- Im LK wird zusätzlich die Gestaltung einer grafischen Benutzungsoberfläche mit einem GUI-Builder thematisiert.
- Die Anwendung folgt dem MVC-Prinzip (Model-View-Controller).
- Die Projektarbeit erfolgt arbeitsteilig, mit Fokus auf Softwareergonomie und Teststrategien.

Q1-II: Anwendungen mit dynamischen, linearen Datenstrukturen

Leitfragen
- Wie verwaltet man Daten nach FIFO oder LIFO?
- Welche Datenstruktur eignet sich für welchen Kontext?

Inhaltliche Schwerpunkte und Verlauf

- Queue, Stack, List
- Operationen und Unterschiede
- Anwendungskontexte

Die Schüler lernen die linearen Datenstrukturen Schlange, Stapel und Liste kennen. Sie analysieren Problemstellungen, modellieren passende Strukturen und implementieren Anwendungen mit den Klassen Queue, Stack und List. Unterschiede und Einsatzmöglichkeiten werden reflektiert.

Kompetenzbereiche (je nach Thema):

- Modellieren (M): Auswahl und Modellierung geeigneter Datenstrukturen.
- Implementieren (I): Implementierung von Operationen und Nutzung generischer Klassen.
- Argumentieren (A): Bewertung der Funktionalität und Effizienz.

- Darstellen und Interpretieren (DI): Visualisierung der Datenstrukturen und Abläufe.

Leistungskurs-Erweiterung:

- Im LK werden eigene Datenstrukturen modelliert und implementiert, nicht nur die Nutzung vorgefertigter Klassen wie Queue, Stack, List.
- Es erfolgt ein Vergleich verschiedener Implementationsstrategien.
- Die Schüler dokumentieren und testen ihre Strukturen systematisch.

Q1-III: Suchen und Sortieren auf linearen Datenstrukturen

Leitfragen

- Wie findet man Daten effizient?
- Welche Sortierverfahren sind besonders leistungsfähig?

Inhaltliche Schwerpunkte und Verlauf

- Lineare und binäre Suche
- Sortieren durch Einfügen, Quicksort
- Effizienzvergleiche

Die Schüler entwickeln und implementieren Such- und Sortierverfahren für Listen und Arrays. Sie analysieren iterative und rekursive Algorithmen und vergleichen deren Effizienz hinsichtlich Laufzeit und Speicherbedarf. Die Verfahren werden grafisch dargestellt und getestet.

Kompetenzbereiche:

- Modellieren (M): Entwicklung und grafische Darstellung von Algorithmen.
- Implementieren (I): Umsetzung und Test von Such- und Sortierverfahren.
- Argumentieren (A): Analyse und Bewertung der Effizienz.
- Darstellen und Interpretieren (DI): Visualisierung und Interpretation algorithmischer Abläufe.

Leistungskurs-Erweiterung:

- Im LK werden Sortierverfahren unterschiedlicher Komplexitätsklassen analysiert.
- Die Schüler entwickeln rekursive Algorithmen wie Quicksort selbstständig.
- Es erfolgt eine grafische Darstellung der rekursiven Abarbeitung und eine Effizienzanalyse mit unterer Schranke.

Q1-IV: Modellierung und Nutzung relationaler Datenbanken

Leitfragen

- Wie speichert und strukturiert man Daten sinnvoll?
- Wie formuliert man effektive Datenbankabfragen?

Inhaltliche Schwerpunkte und Verlauf

- Entity-Relationship-Modellierung
- SQL-Abfragen
- Normalformen

Die Schüler analysieren bestehende Datenbanken und entwickeln eigene Datenbankmodelle. Sie formulieren SQL-Abfragen zur Informationsgewinnung und überprüfen Datenbankschemata auf Konsistenz und Redundanz. Die Normalisierung wird als Qualitätsmerkmal eingeführt.

Kompetenzbereiche (je nach Thema):

- Modellieren (M): Erstellung von ER-Diagrammen und Datenbankschemata.
- Implementieren (I): Umsetzung von Datenbanken und SQL-Abfragen.

- Argumentieren (A): Bewertung von Datenbankmodellen und Abfragen.
- Darstellen und Interpretieren (DI): Visualisierung von Datenbankstrukturen und Ergebnissen.

Leistungskurs-Erweiterung:

- Im LK wird ein vollständiger Softwareentwicklungszyklus mit Datenbankanbindung durchlaufen.
- Die Schüler modellieren, implementieren und testen komplexe Datenbankprojekte (z. B. Quizspiel, Navigationssystem).
- Es wird MVC-basiert gearbeitet, inklusive SQL-Integration in Java.

Q1-V: Sicherheit und Datenschutz in Netzstrukturen

Leitfragen

- Wie schützt man Daten in Netzwerken?
- Welche Verschlüsselungsverfahren gibt es?

Inhaltliche Schwerpunkte und Verlauf

- Netzwerktopologien und Protokolle
- Symmetrische und asymmetrische Verschlüsselung
- Datenschutz und Urheberrecht

Die Schüler analysieren die Struktur von Netzwerken und lernen Verschlüsselungsverfahren kennen. Sie reflektieren Datenschutzaspekte und rechtliche Rahmenbedingungen. Fallbeispiele verdeutlichen die Bedeutung von Sicherheit in der digitalen Welt.

Kompetenzen:

- Argumentieren (A): Bewertung von Sicherheitskonzepten und rechtlichen Aspekten.
- Darstellen und Interpretieren (DI): Visualisierung von Netzstrukturen und Verschlüsselungsprozessen.
- Kommunizieren und Kooperieren (KK): Diskussion und Präsentation von Fallbeispielen.

Leistungskurs-Erweiterung:

- Im LK erfolgt eine tiefere ethische Reflexion (z. B. Verantwortungsethik nach Hans Jonas).
- Die Schüler analysieren komplexe Fallbeispiele (z. B. autonomes Fahren) aus verschiedenen Perspektiven.
- Es werden Kryptoverfahren wie RSA, Diffie-Hellman und digitale Signaturen implementiert und bewertet.

Q2-I: Anwendungen mit dynamischen, nichtlinearen Datenstrukturen

Leitfragen

- Wie lassen sich Daten in Baumstrukturen effizient verwalten?
- Welche Vorteile bieten Suchbäume gegenüber linearen Strukturen?

Inhaltliche Schwerpunkte und Verlauf

- Wie lassen sich Daten in Baumstrukturen effizient verwalten?
- Welche Vorteile bieten Suchbäume gegenüber linearen Strukturen?

Die Schülerinnen und Schüler lernen die Struktur und Funktionsweise von Binärbäumen und Suchbäumen kennen. Sie analysieren Problemstellungen, modellieren Baumstrukturen und implementieren Traversierungs- und Suchalgorithmen. Die rekursive Natur der Bäume wird genutzt, um effiziente Datenverwaltung zu ermöglichen.

Kompetenzen:

- Modellieren (M): Modellierung von Baumstrukturen mit rekursivem Aufbau.
- Implementieren (I): Implementierung von Traversierungs- und Suchalgorithmen.
- Argumentieren (A): Bewertung der Effizienz und Struktur von Baumalgorithmen.
- Darstellen und Interpretieren (DI): Grafische Darstellung von Bäumen und algorithmischen Abläufen.

Leistungskurs-Erweiterung:

- Im LK werden Graphenalgorithmen wie Tiefensuche, Dijkstra und minimale Spannbäume implementiert.
- Die Schüler entwickeln Netzwerkmodelle und simulieren reale Anwendungen (z. B. Versorgungsnetze, soziale Netzwerke).
- Es erfolgt eine Verknüpfung mit GUI und Client-Server-Kommunikation.

Q2-II: Endliche Automaten und formale Sprachen

Leitfragen

- Wie funktionieren Automaten und welche Sprachen erkennen sie?
- Wie hängen Grammatiken und Automaten zusammen?

Inhaltliche Schwerpunkte und Verlauf

- Endliche Automaten (DEA, NEA)
- Reguläre Grammatiken
- Grenzen formaler Sprachen

Die Schüler entwickeln und analysieren endliche Automaten und reguläre Grammatiken. Sie lernen, wie Sprachen formal beschrieben und erkannt werden können. Die Grenzen dieser Systeme werden anhand komplexerer Sprachmuster aufgezeigt. Der Zusammenhang zwischen Grammatik und Automat wird systematisch erarbeitet.

Kompetenzen:

- Modellieren (M): Entwicklung und Modifikation von Automaten und Grammatiken.
- Darstellen und Interpretieren (DI): Darstellung von Automaten als Graphen und Tabellen.
- Argumentieren (A): Analyse der Spracheigenschaften und Grenzen formaler Systeme.

Leistungskurs-Erweiterung:

- Im LK wird ein Parser für eine formale Sprache modelliert und implementiert.
- Die Schüler arbeiten mit nichtdeterministischen Automaten, Kellerautomaten und kontextfreien Grammatiken.
- Es erfolgt eine Verknüpfung mit Compilerbau (Scanner, Parser, Interpreter).

Q2-III: Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit

Leitfragen

- Wie arbeitet ein Computer auf maschinennaher Ebene?
- Welche Probleme sind prinzipiell nicht automatisierbar?

Inhaltliche Schwerpunkte und Verlauf

- Von-Neumann-Architektur
- Maschinennahes Programmieren
- Halteproblem und Grenzen der Berechenbarkeit

Die Schüler lernen die grundlegende Architektur eines Computers kennen und analysieren einfache maschinennahe Programme. Sie setzen sich mit dem Halteproblem auseinander und reflektieren die Grenzen der Automatisierung. Die theoretischen Konzepte werden durch praktische Beispiele veranschaulicht.

Kompetenzen:

- Argumentieren (A): Bewertung der Grenzen von Informatiksystemen.
- Darstellen und Interpretieren (DI): Darstellung der Rechnerarchitektur und Programmabläufe.
- Kommunizieren und Kooperieren (KK): Diskussion ethischer und technischer Fragestellungen.

Grundsätze der fachdidaktischen und fachmethodischen Arbeit

Unter Berücksichtigung der überfachlichen Leitlinien hat die Fachkonferenz Informatik darüber hinaus die folgenden fachdidaktischen und fachmethodischen Grundsätze beschlossen.

- Der Unterricht orientiert sich am aktuellen Stand der Informatik. Dazu beschäftigen sich die Schülerinnen und Schüler auch mit aktuellen Informatiksystemen und deren Weiterentwicklungen.
- Der Unterricht ist problemorientiert, soll von realen Problemen ausgehen, sich auf solche rückbeziehen und knüpft an die Interessen und Erfahrungen der Schülerinnen und Schüler an.
- Der Unterricht ist anschaulich sowie gegenwarts- und zukunftsorientiert und gewinnt dadurch für die Schülerinnen und Schüler an Bedeutsamkeit.
- Der Unterricht ist handlungsorientiert, d. h. projekt- und produktorientiert angelegt.
- Der Unterricht folgt dem Prinzip der Exemplarizität und soll ermöglichen, informatische Strukturen und Gesetzmäßigkeiten in den ausgewählten Problemen und Projekten zu erkennen.
- Der Unterricht fördert vernetzendes Denken und wird deshalb, falls möglich, fach- und lernbereichsübergreifend ggf. auch projektartig angelegt.
- Der Unterricht beinhaltet reale Begegnung sowohl an inner- als auch an außerschulischen Lernorten.
- Im Unterricht werden sowohl für die Schule didaktisch reduzierte als auch reale Informatiksysteme aus der Berufs- und Lebenswelt eingesetzt.

Lehr- und Lernmittel

Da das Gymnasium zurzeit nicht über ein Lehrwerk verfügt, in dem die beschlossenen Unterrichtsvorhaben ausreichend Berücksichtigung finden, arbeiten die Lehrkräfte mit selbst zusammengestellten Materialien. Diese werden den Schülern über die Schulplattform (idR über OneNote als Teil von Microsoft 365) bereitgestellt, so dass sie auch rückwirkend und unterrichtsbegleitend jederzeit darauf Zugriff haben. Somit können auch versäumte Unterrichtsstunden nachgearbeitet und sich auf die Kursarbeiten vorbereitet werden.

Qualitätssicherung und Evaluation

Maßnahmen der fachlichen Qualitätssicherung

Die Fachkonferenz überprüft kontinuierlich, inwieweit die im schulinternen Lehrplan vereinbarten Maßnahmen zum Erreichen der im Kernlehrplan vorgegebenen Ziele geeignet sind. Dazu dienen beispielsweise auch der regelmäßige Austausch sowie die gemeinsame Konzeption von Unterrichtsmaterialien, welche hierdurch mehrfach erprobt und bezüglich ihrer Wirksamkeit beurteilt werden.

Kolleginnen und Kollegen der Fachschaft nehmen regelmäßig an Fortbildungen teil, um fachliches Wissen zu aktualisieren und pädagogische sowie didaktische Handlungsalternativen zu entwickeln. Zudem werden die Erkenntnisse und Materialien aus fachdidaktischen Fortbildungen und Implementationen zeitnah in der Fachgruppe vorgestellt und für alle verfügbar gemacht.

Feedback von Schülerinnen und Schülern wird als wichtige Informationsquelle zur Qualitätsentwicklung des Unterrichts angesehen. Sie sollen deshalb Gelegenheit bekommen, die Qualität des Unterrichts zu evaluieren. Dafür kann das Online-Angebot SEFU (Schüler als Experten für Unterricht) genutzt werden (https://www.sefu-online.de, Datum des letzten Zugriffs: März 2024) oder auch das Tool Forms der Microsoft 365 Schulcloud.

Überarbeitungs- und Planungsprozess

Eine Evaluation erfolgt jährlich. In der Fachkonferenz zu Schuljahresbeginn werden die Erfahrungen des vorangehenden Schuljahres ausgewertet und diskutiert sowie eventuell notwendige Konsequenzen formuliert. Die vorliegende Checkliste wird als Instrument einer solchen Bilanzierung genutzt. Nach der jährlichen Evaluation werden Änderungsvorschläge für den schulinternen Lehrplan eingearbeitet. Insbesondere findet eine Verständigung über alternative Materialien, Kontexte und die Zeitkontingente der einzelnen Unterrichtsvorhaben statt.