

KURSPLAN

FRONTENDUTVECKLING

Denna kurs fokuserar på frontendutveckling med JavaScript, inklusive skapandet av dynamiska och interaktiva webbapplikationer. Studerande lär sig att hantera DOM, integrera API:er och implementera tillgänglighetsanpassningar enligt WCAG. Kursen täcker också grundläggande UX/UI-principer, prestandaoptimering och säkerhet för att skapa användarvänliga och effektiva gränssnitt. Efter en inledande inlärningsfas genomför studerande projekt där de praktiserar sina färdigheter i en arbetsliknande miljö.

Lärandemål

Vid avslutad kurs med godkända resultat, kan den studerande:

Kunskaper

1. Beskriva hur eventhantering i JavaScript används för att skapa interaktiva webbapplikationer.
2. Förklara hur API:er används för att hämta och manipulera data i webbapplikationer.
3. Beskriva principerna för att utveckla modulära och återanvändbara JavaScript-komponenter.
4. Redogöra för WCAG-standarder och hur dessa tillämpas för att skapa tillgängliga gränssnitt.
5. Beskriva UX- och UI-principer och hur dessa används för att skapa användarvänliga gränssnitt.
6. Förklara grundläggande säkerhetsprinciper för att skydda användardata i webbapplikationer.
7. Förklara grunderna i testning av frontendapplikationer, inklusive användning av testverktyg som Jest och Cypress för enhetstester och end-to-end-tester.
8. Förklara hur AI-verktyg kan användas för att generera, analysera och förbättra JavaScript-kod, inklusive prestandaoptimering och identifiering av kodlukt.
9. Förklara hur agila metoder, såsom Scrum och Kanban, kan användas för att strukturera och effektivisera frontendutvecklingsprocessen.
10. Förklara hur dokumentation kan användas för att beskriva JavaScript-funktioner, komponenter och API-anrop, samt vikten av konsekvent struktur i dokumentationen.

Färdigheter

1. Utföra enklare DOM-manipulation med JavaScript.
2. Skapa interaktiva webbapplikationer med JavaScript och DOM-manipulation.
3. Integrera API:er för att visa och manipulera data i webbapplikationer.



4. Implementera tillgänglighetsanpassningar enligt WCAG-standarder.
5. Designa och utveckla användarvänliga gränssnitt enligt UX/UI-principer.
6. Använda AI-baserade verktyg för att analysera och optimera JavaScript-kod, inklusive identifiering av kodlukt och prestandaförbättringar.
7. Använda en agil arbetsmetodik för att planera och genomföra frontenduppgifter, inklusive att bryta ner stories, arbeta i sprintar och hantera uppgifter i ett Kanban-board.
8. Skriva och genomföra automatiserade enhetstester för JavaScript-funktioner och UI-komponenter med hjälp av Jest.
9. Dokumentera en frontendkodbas genom att skriva detaljerade JSDoc-kommentarer och instruktioner för hur man använder och testar JavaScript-funktioner och UI-komponenter.

Kunskapskontroll

Summativ bedömning

Summativ bedömning utvärderar studerandes kunskaper och färdigheter retroaktivt för att sammanfatta vad den studerande kan vid examinationstillfället. Den summativa bedömningen täcker samtliga kursmål och sker vid följande avgränsade examinationstillfällen:

- Skriftligt kunskapstest – En (1) individuell prövning av teoretiska kunskaper.
- Skriftlig reflektion – En (1) individuell redogörelse för egna lösningar och tillvägagångssätt.
- Projektinlämning – En (1) gruppinlämning där varje studerandes individuella insats bedöms.

Formativ bedömning

Formativ bedömning utvärderar studerandes kunskaper och färdigheter löpande under kursens interaktiva lärandemoment. Syftet är att effektivisera bedömningen och samtidigt stödja lärandet. Den formativa bedömningen sker i samband med:

- Workshops – Interaktiva undervisningstillfällen där klassen och utbildaren övar tillsammans.
- Checkpoints – Regelbundna avstämningar där studerande presenterar sina projekt och får direkt återkoppling.

Formativ bedömning är ett komplement till summativ bedömning och tillämpas enbart till fördel för studerande. Studerande kan uppnå kursmål redan innan de summativa examinationerna genom formativ bedömning, men ett underkänt resultat kan aldrig enbart baseras på denna bedömningsform.

Principer för bedömning och betygskriterier



- **Icke godkänt (IG):**
Alla lärandemål är inte uppfyllda.
 - **Godkänt (G):**
Alla lärandemål är uppfyllda på en tillfredsställande nivå.
 - **Väl godkänt (VG):**
Alla lärandemål är uppfyllda på en tillfredsställande nivå. Utöver ovan uppvisar den studerande:
 - Hög precision i utförande av enklare DOM-manipulation med JavaScript.
 - Stor skicklighet vid skapande av interaktiva webbapplikationer med JavaScript och DOM-manipulation.
 - Hög precision och korrekthet vid integrering av API:er för att visa och manipulera data i webbapplikationer.
-

Rätt till särskilt pedagogiskt stöd och komplettering

Vid behov av särskilt pedagogiskt stöd kan följande anpassningar göras vid uppvisande av intyg:

- Förlängd tid vid examinationstillfället.
- Alternativa examinationsformer, exempelvis skriftliga istället för muntliga moment.
- Utökat stöd för att underlätta inläringen.

Komplettering och omprov:

- Studerande har rätt till en (1) omexamination inom sex (6) månader från det ordinarie examinationstillfället. Tid och form fastställs av utbildningsanordnaren och utbildaren och meddelas till studerande, som förväntas följa dessa beslut.
 - Om omprovet inte genomförs eller inte godkänns, kan inget betyg sättas i kursen. Studerande kan dock fortsätta med efterföljande kurser, förutsatt att den missade kursen inte utgör ett förkunskapskrav för dessa.
-

