



LUNDS UNIVERSITET

Ekonomihögskolan

Institutionen för informatik

AI-verktygs inverkan på utvecklare

En kvalitativ studie om utvecklares erfarenhet och acceptans av AI-verktyg inom programmeringskurser

Kandidatuppsats 15 hp, kurs SYSK16 i Informationssystem

Författare: Erik Lycke
Carl Berg

Handledare: Paul Pierce

Rättande lärare: Odd Steen
Umberto Fiaccadori

AI-verktygs inverkan på utvecklare: En kvalitativ studie om utvecklarens erfarenhet och acceptans av AI-verktyg inom programmeringskurser

ENGELSK TITEL: AI-tools and their impact on developers: A qualitative study on developers experience and acceptance of AI-tools in programming courses.

FÖRFATTARE: Erik Lycke och Carl Berg

UTGIVARE: Institutionen för informatik, Ekonomihögskolan, Lunds universitet

EXAMINATOR: Osama Mansour, Docent

FRAMLAGD: maj, 2024

DOKUMENTTYP: Kandidatuppsats

ANTAL SIDOR: 101

NYCKELORD: Artificiell Intelligens, Erfarenhet, Acceptans, Programmering

SAMMANFATTNING (MAX. 200 ORD):

I takt med att flera prominenta AI-verktyg har visat sig effektiva i programmeringssyfte har frågan kring utvecklarens erfarenhet och acceptans av dessa AI-verktyg ställts. Studien syftar till att utforska utvecklarens erfarenheter och acceptans av AI-verktyg i programmeringskurser med fokus på faktorerna erfarenhet, förväntad prestanda förväntad ansträngning, socialt inflytande, underlättande förhållanden och framtid. Genom semistrukturerade intervjuer och en tematisk analys baserad på UTAUT-modellen visar resultaten att det finns en bred acceptans till användandet av AI-verktyg inom programmering där erfarenhet spelar en avgörande roll för användarnas acceptans. Förväntad prestanda, förväntad ansträngning, socialt inflytande och underlättande medel är också väsentliga faktorer för användningen av dessa verktyg. Framtidsutsikterna för AI-assisterad programmering diskuteras med en blandning av optimism och oro. Denna studie bidrar till en fördjupad förståelse för användningen av AI-verktyg inom

mjukvaruutveckling i en utbildningskontext och pekar på vikten av att beakta användarnas erfarenheter och förväntningar för en framgångsrik implementering av dessa verktyg.

Innehåll

AI-verktygs inverkan på utvecklare: En kvalitativ studie om utvecklarens erfarenhet och acceptans av AI-verktyg inom programmeringskurser.....	I
Innehåll.....	IV
Figurer.....	VII
Tabeller.....	VII
Termer och förkortningar.....	2
1. Introduktion.....	3
1.1 Bakgrund.....	3
1.2 Problemområde.....	4
1.3 Forskningsfråga.....	4
1.4 Syfte.....	5
1.5 Avgränsningar.....	5
2. Litteraturgenomgång.....	6
2.1 Artificiell Intelligens.....	6
2.1.1 Definition och bakgrund.....	6
2.1.2 Maskininlärning.....	7
2.1.3 Naturlig språkbehandling.....	7
2.2 Generativ-AI.....	8
2.2.1 Adoptionen av generativ-AI i mjukvaruutveckling.....	8
2.3 AI-verktyg.....	9
2.3.1 ChatGPT.....	10
2.3.2 GitHub Copilot.....	11
2.4 Mjukvaruutveckling.....	12
2.4.1 Mjukvaruutveckling bakgrund.....	12
2.4.2 AI-assisterad mjukvaruutveckling.....	12
2.4.3 Människa-AI samarbete.....	13
2.4.4 AI-augmentation.....	13
2.5 Unified Theory of Acceptance and use of Technology.....	14
2.5.1 Ramverkets delar.....	14

2.5.2	Hur UTAUT tillämpas	16
2.5.3	Kritik mot UTAUT och eventuella utbyggnader av modellen.....	16
2.5.4	Litteratursammanfattning	16
3	Metod	18
3.1	Metodval.....	18
3.2	Forskningsfilosofi, Kvalitativ metod	18
3.2.1	Kritik mot metoden	18
3.2.2	Valet av intervjuer som insamlingsmetod.....	19
3.3	Datainsamling	20
3.3.1	Litteraturkapitel	20
3.3.2	Utformande & Utförande av Intervju.....	21
3.3.3	Motivering av Intervjuguide.....	22
3.3.4	Urval av Respondenter	23
3.3.5	Transkribering	24
3.4	Analys.....	24
3.4.1	Förberedande analys.....	24
3.4.2	Innehållsanalys	24
3.5	Etik.....	25
3.5.1	Etiska övervägande	25
3.6	Datakvalitet.....	26
3.6.1	Validitet & reliabilitet	26
4	Empiri	28
4.1	Erfarenhet	28
4.1.1	Användning av AI inom programmering.....	28
4.2	Acceptans.....	29
4.2.1	Förväntad prestanda	29
4.2.2	Förväntad ansträngning	31
4.2.3	Social påverkan	32
4.2.4	Underlättande förhållande	34
4.6	Framtiden för Generativ-AI & AI förstärkning.....	35
5	Diskussion.....	37
5.1	Erfarenhet	37
5.2	Acceptans.....	38

5.2.1	Förväntad prestanda	38
5.2.2	Förväntad ansträngning	39
5.2.3	Socialt inflytande.....	40
5.2.4	Underlättande förhållande	41
5.3	Framtid.....	42
6	Slutsats	44
6.1	Kritisk inblick	45
6.1.1	Studiens skala & urval av respondenter	45
6.1.2	Författarnas Bias	45
6.1.3	Empiriskt material	45
6.2	Förslag till vidare forskning	45
Appendix A – intervju R1		47
Appendix B – intervju R2		Error! Bookmark not defined.
Appendix C – intervju R3		Error! Bookmark not defined.
Appendix D – Intervju R4.....		Error! Bookmark not defined.
Appendix E – intervju R5.....		Error! Bookmark not defined.
Appendix F – Undersökningsmodell.....		Error! Bookmark not defined.
Appendix G – Consent form		Error! Bookmark not defined.
Appendix H - AI-bidragsredogörelse		48
7	Referenser	51

Figurer

Figur 2.1 Släppta LLM:s (Naveed et al. 2023).....	11.
Figur 2.2 UTAUT ramverket. (Venkatesh et al. 2003)	22.

Tabeller

Tabell 1.1: Termer och förkortningar.....	8.
Tabell 2.1 Litteratursammanfattning.....	23.
Tabell 3.1 Litteratursökningstabell.....	28.
Tabell 3.2 Intervjuobjekt.....	30.
Tabell 3.3 Kodningstabell.....	32.

Termer och förkortningar

1.1 Termer och förkortningstabell: Vanligt förekommande termer och förkortningar med respektive definition.

Term/förkortning:	Definition:
AI	Artificiell Intelligens: vetenskapligt fält som syftar till att skapa maskiner som kan utföra uppgifter som kräver mänsklig intelligens (Russel & Norvig, 2020).
IoT	Sakernas Internet: alla de enheter som via inbyggda sensorer eller datorer är uppkopplade mot internet (Dwivedi et al., 2021)
LLM	Stora Språkmodeller: djupinlärnings (DL) modeller, tränade för att förstå och generera naturligt språk (Cambria & White, 2014)
ML	Maskininläring: en kategori inom Artificiell Intelligens (AI) av algoritmer och statistiska metoder som låter datorsystem utföra uppgifter utan att ha blivit explicit programmerade (Mahesh, 2020)
NLP	Naturlig språkbehandling: en kategori inom maskininläring (ML) som fokuserar på interaktionen mellan datorer och människor via naturligt språk (Agarwal, 2019).
Generativ-AI	Generativ Artificiell Intelligens: ny form av artificiell intelligens som använder maskininlärningsalgoritmer och modeller för att generera nytt innehåll (Nhavkar & Goel, 2023)
UTAUT	Unified theory of acceptance and use of technology: teorin om acceptans och användning av teknologi är en teknologiacceptansmodell (Venkatesh et., 2003).

1. Introduktion

Introduktions avsnittet som följer har avsikten att presentera bakgrunden till uppsatsämnet och presentera problemområdet, forskningsfrågan samt syfte och avgränsningar för studien.

1.1 Bakgrund

I takt med den snabba utveckling inom Artificiell Intelligens (AI), framträder en skiftande arbetsmarknad där många jobb riskerar att ersättas av AI-baserad teknologi (Dwivedi et al. 2021). Detta skifte på jobbmarknaden speglar dock en historisk kontinuitet snarare än ett nytt fenomen hävdar Dwivedi et al. (2021). Redan under den första industriella revolutionen kunde man se ett betydande skifte från jordbrukssamhälle till ett industrisamhälle, där mekaniseringen, drivet av ångkraft, gradvis ersatte människans fysiska arbete med maskinens effektivitet (Blom et al. 2014). Nu menar flera forskare på att den fjärde industriella revolutionen anlönt. Drivet av integreringen av digitala teknologier som Sakernas internet (IoT), artificiell intelligens (AI), maskinlärning (ML), cyberfysiska system i alla aspekter av våra liv, och maskinintelligens som optimerar, förstärker eller automatiserar komplexa teknologier som generativ-AI för effektiva lösningar (Blom et al. 2014; Dwivedi et al. 2021).

AI är ett modeord som länge har fascinerat och orsakat oro i lika mått, ett begrepp som nästan känns lika hemma i vetenskapliga journaler som i science-fiction berättelser. Ända sedan AI:s början har dess definitionen skiftat, men Brynjolfsson, Li, Raymond, (2023) definierar AI som ett samlingsbegrepp vilket avser system som uppvisar intelligent beteende, såsom inlärning, problemlösning och resonemang.

Enligt studierna av World Economic Forum (WEF) som Dwivedi et al. (2021) lyfter i sin studie, förutspås AI:s förmåga påverka omkring 20% av de befintliga jobben i England och upp till 26% i större framväxande ekonomier. Trots att AI potentiellt kan ersätta vissa jobb, visar tidigare forskning av Brynjolfsson & McAfee (2017) att AI sällan ersätter hela processer eller jobb. Istället fungerar teknologin oftare som ett komplement eller verktyg som underlättar mänskliga aktiviteter (Brynjolfsson & McAfee, 2017). Därför menar Brynjolfsson & McAfee (2017) att AI är som mest effektivt när det används för att automatisera specifika steg i en process, vilket låter människor fokusera på mer komplexa, kreativa och kognitivt orienterade roller som stöttar AI teknologin (Brynjolfsson och McAfee 2017; Dwivedi et al. 2021). Idén att människor bör arbeta sida vid sida med AI i stället för att låta den ersätta hela processer betonar Dwivedi et al (2021) som en viktig punkt och efterlyser mer forskning kring. Vidare så betonar Brynjolfsson, Li, Raymond, (2023) i sin forskning att det krävs mer forskning på generativ-AI:s påverkan på arbetsprocesser för att förstå potentiellt långsiktiga effekter.

Tidigare forskning utav Naveed et al. (2023) pekar på att framväxten av AI-baserade verktyg har lett till signifikanta förändringar i mjukvaruutvecklingsprocessen samt har haft en

inverkan på utvecklarens arbetsprocess (Naveed et al. 2023; Kuhail et al. 2024; Zhang et al. 2023). Fortsättningsvis så lyfter Özpolat et al. (2023) att när AI-verktyg används för att automatisera specifika steg i utvecklingsprocessen kan utvecklare bygga system snabbare, smartare och mer effektivt vilket förbättrar den traditionella utvecklingsprocessen. Något studierna av Zhang et al. (2023) stödjer, där användningen av AI-verktyget Github Copilot anges ha lett till snabbare kodning av högre kvalitet.

1.2 Problemområde

I introduktionen noterades det, baserat på Dwivedi et al. (2021) och Brynjolfsson och McAfee (2017), hur generativ-AI kan ersätta eller komplettera mänskligt arbete. Detta skapar en dubbel dynamik i arbetsmarknaden som behöver ytterligare utforskning, särskilt inom områden där generativ-AI inte bara stödjer utan potentiellt kan ersätta mänskliga färdigheter. Som tidigare framhävts av Zhang et al. (2023), har integreringen av AI-verktyg i mjukvaruutveckling revolutionerat den traditionella mjukvaruutvecklingsprocessen. Denna transformation innebär dock inte bara effektivitet utan också nya utmaningar i att balansera mellan automatisering och mänskligt omdöme, vilket utgör kärnan i detta problemområde.

Precis som med alla teknologiska övergångar medför generativ-AI både betydande möjligheter och utmaningar. Chui et al. (2023) menar på att generativ-AI inom mjukvaruutveckling kan bidra med effektivitet till ett värde av cirka 485 miljarder USD inom IT, respektive 414 miljarder USD i produktionsutveckling. Användningen av generativ-AI, som har potential att bidra med miljarder dollar till utvecklingsindustrin, medför också betydande risker som bör adresseras. Naveed et al. (2023) lyfter bland annat olika typer av AI hallucinationer som kan medföra missledande och förvirring för användaren. Vidare finner Dakhel et al. (2023) att AI-verktyg som Github Copilot genererar kod snabbt, men att koden kan vara buggig och ej replikerbar.

Detta skifte exemplifieras idag av generativa AI-verktyg som OpenAI:s ChatGPT och Githubs github Copilot, två AI-baserade språkmodeller som har används för par-programmering (Zhang et al., 2023; Özpolat et al., 2023). Enligt studien av Özpolat et al. (2023) och Zhang et al. (2023) har AI-verktyg, där bland annat Copilot och ChatGPT, blivit kritiska för att driva effektivitet och innovation inom mjukvaruutveckling och antyder en banbrytande utveckling i mjukvaruutvecklingsprocessen. Även om dessa verktyg hyllats för sin förmåga att effektivisera, måste utvecklare också navigera sig genom potentiella risker som AI-hallucinationer och icke replikerbar kod (Dakehl et al, 2023; Naveed et al, 2023). Således väcker detta frågan, Vad är utvecklarens erfarenhet och acceptans av AI-verktyg inom mjukvaruutveckling?

1.3 Forskningsfråga

”Vad är utvecklarens erfarenhet och acceptans av AI-verktyg för mjukvaruutvecklingsprocesser inom ramen för programmeringskurser”

1.4 Syfte

Antalet AI-verktyg som har förmågan att assistera utvecklare vid mjukvaruutveckling har ökat stadigt (Naveed et al. 2023). Samtidigt tyder forskningen på att förmågan hos AI-verktyg ökar och därmed ökar graden av automatisering som är möjlig i mjukvaruutvecklingsprocessen (Naveed et al. 2023; Kuhail et al., 2024; Zhang et al. 2023).

Författarna argumenterar därför i denna studie för att det behövs en djupare förståelse för att fastställa hur utvecklare för närvarande upplever denna förändring och rörelse mot ytterligare automatisering inom mjukvaruutveckling, oavsett om den i nuläget är begränsad till assisterande verktyg eller autokompletterande verktyg. Uppsatsen syftar därför att utforska och analysera utvecklarens erfarenheter och acceptans av AI-verktyg i mjukvaruutveckling inom ramen för programmeringskurser.

1.5 Avgränsningar

Studien fokuserar på användningen av AI-verktyg i mjukvaruutveckling inom ramen för programmeringskurser, men har avgränsat till AI-verktygen GitHub Copilot och ChatGPT eftersom det är dessa AI-verktygen som används primärt av respondenterna. De specifika AI-verktygen har även valts på grund av deras framstående roll i den aktuella teknikutvecklingen och deras specifika tillämpningar inom kodgenerering och programmeringsassistans, vilket är direkt relevant för forskningsfrågan. För att säkerställa en hanterbar och fokuserad undersökning, avgränsas studiepopulationen till lärare, studenter och labbassistenter vid Institutionen för Informatik vid Lunds universitet samt studenter på Datasektionen vid Lunds Tekniska Högskola. Denna geografiska och demografiska avgränsning möjliggör detaljerade studier av specifika grupper som regelbundet interagerar med AI-verktyg.

2 Litteraturgenomgång

Litteraturkapitlet kommer att presentera tidigare forskning inom ämnet, utforska relevant litteratur som ger en grundlig förståelse för ämnet samt presentera ramverket som används för att analysera empiriinsamlingen.

2.1 Artificiell Intelligens

2.1.1 Definition och bakgrund

Artificiell intelligens (AI) representerar en av de mest avancerade teknologierna som finns tillgängliga idag, med betydande potential att förbättra olika operationer och ge betydande fördelar för flera sektorer (Russel & Norvig, 2021). Ämnet som har sin grund i flera olika discipliner, där bland annat datavetenskap och kognitiv psykologi. är ett omfattande och komplext område, vilket har gjort det svårt att definiera (Russel & Norvig, 2020).

Enligt Russel & Norvig (2020) & Mitchell (2021) har det inte alltid funnits en entydig definition av ämnet, vilket enligt Mitchell (2021) tidigare berott på en begränsad förståelse för intelligensens verkliga natur och komplexitet. Däremot menar Russel & Norvig (2020) på att trots en viss otydlighet kring den exakta definitionen så har det generella målet med AI kunnat definieras som ett vetenskapligt fält vilket syftar till att skapa maskiner som kan utföra uppgifter som kräver mänsklig intelligens. Denna definition går i linje med definitionen som organisationen för ekonomiskt samarbete och utveckling (OECD), fastställde i maj 2019. OECD skapade ett set principer för AI med syftet att främja pålitlig och effektiv AI-utveckling. Principerna som syftade till att etablera en global standardiserad definition av AI löd såhär:

”ett maskinbaserat system som, för explicita eller implicita mål, härleder från den inmatning den får, hur man genererar utdata såsom förutsägelser, innehåll, rekommendationer eller beslut som kan påverka fysiska eller virtuella miljöer. Olika AI-system varierar i sina nivåer av autonomi och anpassningsförmåga efter implementering” (OECD, 2024, n.p.).

Trots en globalt standardiserad definition av AI så är det viktig att förstå bakgrunden och komponenterna bakom paraplytermen AI. Begreppen AI, maskininlärning och djupinlärning, som ofta används synonymt, representerar faktiskt olika kategorier av AI och utgör delmängder av varandra (IBM, 2024). AI innebär den övergripande idén att skapa intelligenta maskiner som kan utföra uppgifter som traditionellt sett kräver mänsklig intelligens. Maskininlärning involverar träning av datorer för att känna igen mönster i data och göra förutsägelser eller beslut baserat på denna data, medan djupinlärning utnyttjar neurala nätverk med flera lager för att analysera och bearbeta komplexa data (IBM, 2024).

2.1.2 Maskininlärning

Maskininlärning (ML), en kategori inom AI, definieras av Mahesh (2020) som den vetenskapliga studien av algoritmer och statistiska metoder vilket låter datorsystem utföra uppgifter utan att ha blivit explicit programmerade. ML innefattar utvecklingen av maskininlärningsalgoritmer som tillåter datorer att bearbeta data och utifrån erfarenhet förbättra sin prestanda (Jordan & Mitchell, 2015). Denna förmåga att lära sig från erfarenheter och förbättra prestanda över tid gör ML till en central del av många AI-verktyg menar Jordan & Mitchell (2015). Dock belyser Mahesh (2020) att inlärning i maskininlärningssammanhang inte är synonymt med mänsklig medvetenhet utan snarare en teknisk process för att identifiera mönster och regelbundenheter i data.

Inom ML finns olika typer av maskininlärningsalgoritmer, såsom övervakad inlärning, oövervakad inlärning, semi-övervakad inlärning och förstärkningsinlärning. Maskininlärningsalgoritmer finns integrerat i mycket av den tekniken vi använder dagligen, från sökmotorn Google där inlärningsalgoritmer har blivit tränade till att ranka websidor, till intelligenta chattbotar (Mahesh, 2020 & Sarker, 2021). Algoritmernas främsta fördel menar Mahesh (2020) är dess förmåga att kunna lära sig en uppgift och sedan kunna utföra den automatiskt och mer effektivt. På grund av detta är maskininlärningsalgoritmer en viktig komponent i AI-verktyg (Mahesh, 2020).

I takt med att maskininlärningsalgoritmer ML har fått tillgång till större och mer varierade datamängder att tränas på, har deras användning i AI-verktyg förstärkts avsevärt (Roumeliotis & Tselikas, 2023). Enligt Roumeliotis & Tselikas (2023) har den kontinuerliga förbättringen av dessa modeller, genom både övervakad och oövervakad inlärning, gjort det möjligt för AI-verktyg att producera svar som inte bara är relevanta och sammanhängande utan också anpassningsbara till olika användarkontexter. Enligt Zhang et al. (2023) är detta särskilt framträdande inom AI-assisterad programmering, där språkmodeller som GitHub Copilot använder sig av avancerade ML-metoder för att effektivisera och förbättra programmeringsprocesser.

2.1.3 Naturlig språkbehandling

Naturlig språkbehandling (NLP) definieras som en kategori inom maskininlärning (ML) som fokuserar på interaktionen mellan datorer och människor via naturligt språk (Agarwal & Saxena, 2019). Enligt Cambria & White (2014) syftar NLP till att automatiskt analysera och representera mänskligt språk, på så sätt härleda mening från mänskligt språk på ett intelligent och användbart sätt (Agarwal & Saxena, 2019). Tekniken krediteras därför med att ha gett datorer förmågan att organisera och strukturera kunskap för att kunna utföra uppgifter såsom översättning, sentimentanalys, taligenkänning och ämnessegmentering (Agarwal & Saxena, 2019; Cambria & White, 2014). NLP:s förmåga att analysera stora mängder textdata och utvinna relevant information spelar en stor roll i dagens AI-verktyg (Agarwal & Saxena, 2019). I Roumeliotis & Tselikas (2023) artikel framgår det att NLP är en grundläggande teknik bakom stora språkmodeller som ChatGPT, och har gjort det möjligt för språkmodellen att förstå och generera mänskligt språk. Ytterligare så har NLP försett ChatGPT med förmågan att

hantera komplexa språkuppgifter som dialoggenerering och textkomplettering, vilket spelar en central roll i att förbättra kommunikationen mellan människor och AI-verktyg.

2.2 Generativ-AI

Generativ Artificiell Intelligens (Generativ-AI) är en ny form av AI som använder maskininlärningsalgoritmer och modeller för att generera nytt innehåll (Nhavkar & Goel, 2023). Till skillnad från traditionella AI-modeller som enligt Nhavkar & Goel (2023) primärt fokuserar på klassificerings och igenkänningsuppgifter, syftar generativa-AI modeller till att skapa nya data i form av text, bilder, ljud, kod och mera utifrån stora mängder träningsdata (Brynjolfsson, Li & Raymond, 2023). Teknologin, som enligt Nah, Zheng, Cai, Siau, Chen, (2023) är en gren av oövervakad maskininlärning har exemplifierats i stora språkmodeller som ChatGPT och Github Copilot (Bandi, Adapa & Kuchi, 2023).

2.2.1 Adoptionen av generativ-AI i mjukvaruutveckling

Inom IT-sektorn, som Nhavkar (2023) belyser i sin artikel, har generativ-AI visat sig kunna förändra arbetsflöden genom att automatisera och förbättra utvecklingsprocesser. Denna teknologi bidrar till en ökad effektivitet menar han och kan hjälpa programmerare med både kodgenerering och felsökning, vilket reducerar utvecklingstiden och ökar produktiviteten (Nhavkar, 2023). Fortsättningsvis diskuterar Bandi et al. (2023) hur generativ-AI har transformerat utvecklingen av mjukvara och system genom att introducera avancerade modeller som kan generera kod och hantera komplex datahantering. Deras forskning understryker även hur viktigt det är med noggrann validering och kontinuerlig övervakning av dessa AI-verktyg för att säkerställa deras effektivitet och tillförlitlighet i verkliga tillämpningar som ChatGPT (Bandi et al. 2023). Denna ökade produktivitet stöds av Brynjolfsson, Li och Raymond (2023) som noterar i sin forskning att tillgång till generativa-AI verktyg ökade produktiviteten för kundtjänstmedarbetare med 14% i genomsnitt. Brynjolfsson, Li och Raymond (2023) noterade även att ökningen hade större effekt hos mindre kvalificerade medarbetare och påstås ha sänkt erfarenhetskurvan. Van Slyke, Johnsson och Sarabadani, (2023) belyser däremot att adoptionen av Generativ-AI inom mjukvaruutveckling ha lett till förändrade krav på arbetsmarknaden. Generativa AI-verktyg kan automatisera vissa uppgifter som tidigare utfördes av människor, vilket höjer ribban för vilken typ av arbete som förväntas av framtida utvecklare. Författarna diskuterar hur detta skifte kan resultera i att jobben kräver en högre grad av teknisk kompetens och problemlösningsförmåga (Van Slyke et al. 2023).

Adoptionen av generativ-AI har belyst teknologins förmåga att i vissa fall producera felaktig eller påhittad information, vilket har blivit känt som "AI-hallucinerings" enligt McAfee, Rock, & Brynjolfsson (2023). Trots dessa utmaningar har detta inte hindrat framstegen inom området. Enligt undersökningar från det amerikanska företaget Salesforce anser en betydande majoritet av över 500 seniora IT-ledare att generativ-AI är en prioritet för deras företag inom de kommande 18 månaderna. Av dessa ansåg en tredjedel att det var den högsta prioriteringen

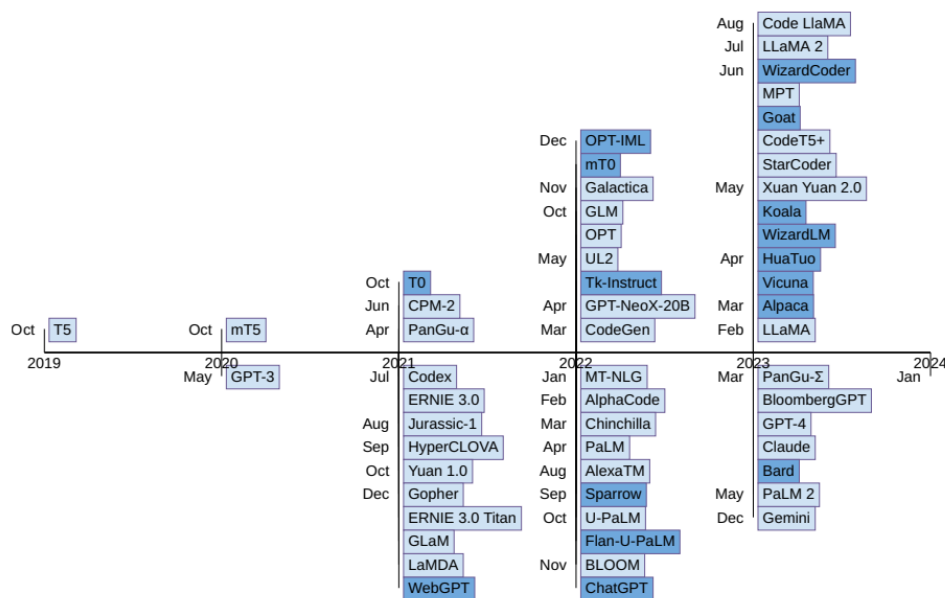
(Salesforce, 2023). Majoriteten (57%) av de tillfrågade IT-ledarna betraktar generativ-AI som en "game changer" och tror att det har potentialen att optimera och öka effektiviteten (Salesforce, 2023).

Trots det stora intresset för generativ-AI och viljan att integrera teknologin i verksamheten, menar Salesforce (2023) att över hälften av dagens ledare inom IT-branschen inser att åtgärder måste vidtas för att använda den på ett framgångsrikt och ansvarsfullt sätt. Utmaningarna sträcker sig bortom en enkel känsla av teknologisk överoptimism. En majoritet av IT-ledarna i undersökningen är skeptiska till de etiska aspekterna av generativ-AI. Enligt samma studie ansåg 59% att outputen från generativ-AI ofta är felaktig och 63% rapporterade en uppfattad partiskhet i outputen, där fenomenet AI-hallucinationer inkluderades. Trots detta kunde även skeptikerna till generativ-AI i studien erkänna vissa fördelar med teknologin. Hela 80% av de som ansåg att teknologin är överdriven eller "over-hyped" tror ändå att generativ-AI kan bidra till en förbättring. Detta förstärker Brynjolfsson et al. (2023) genom att argumentera för att trots vissa tekniska begränsningar och risken för missvisande information kan generativ-AI erbjuda betydande fördelar genom att förbättra effektiviteten och kvaliteten på arbetsutförandet över olika yrkesroller och industrier.

2.3 AI-verktyg

Stora språkmodeller (LLMs) för att automatiskt komplettera källkod blir alltmer populära inom programutveckling (Zhang et al. 2023). Numera utrustas LLM:s med avancerade förmågor för bearbetning av NLP, och ML-modeller har tillämpats på källkodstexter i en mängd nya AI-verktyg som stöder mjukvaruutvecklingsprocessen (Zhang et al. 2023). Detta möjliggör användningen av LLM:s för att generera kod i allmänna programmeringsspråk. På senare tid har denna typ av AI-verktyg för kodgenerering blivit alltmer framträdande, med generativa förtränade språkmodeller (GPT) som tränats på stora mängder kod för att erbjuda lämpliga förslag på kodkomplettering när programmerare skriver kod (Zhang et al. 2023).

AI-verktyg för assisterad programmering har revolutionerat hur man ser på mjukvaruutvecklingsprocessen med sin förmåga att automatisera testning och felsökning, analysera, optimera och generera kod (Kuhail et al. 2024). Utvecklingen av dessa verktyg har bidragit till att underlätta mjukvaruutvecklingsprocessen genom att delvis automatisera monotona aktiviteter och ge förslag för att förbättra programmerares produktivitet och effektivitet (Kuhail et al. 2024). Teknologiska framstegen i kärnområden vilket LLM tekniken bygger på har lett till utvecklandet av flera AI-verktyg av några av världens ledande tech-bolag. År 2022 såg vi en ökning i lanserandet av LLM produkter där diagrammet i **Figur 2.1** visar på den stora ökningen samt trenden mot instruktions baserade modeller som är "open-source" (Naveed et al. 2023).



Figur 2.1: Släppta LLM i kronologisk ordning. Övre delen visar modeller som är "open-source" medan nedre delen visar "closed-source" modeller. Ljusblåa LLM representerar "pre-trained" medan mörkblåa representerar "instruction-based" (Naveed et al., 2023).

En av det största farhågorna som uppmärksammats med AI-verktyg är AI-hallucinationer (Naveed et al. 2023; Shen et al. 2023). Fenomenet beskrivs utav Shen et al. (2023) som en hallucinationseffekt där språkmodeller genererar svar som verkar trovärdiga men som i själva verket är felaktiga eller baserade på icke-existerande fakta. Detta problem tas upp som en betydande utmaning vid användandet av AI-verktyg i tidigare forskning (Naveed et al. 2023; Shen et al. 2023). Både Naveed et al. (2023) och Shen et al. (2023) belyser vikten av att vidareutveckla och förbättra modellernas noggrannhet och tillförlitlighet och menar på att det är avgörande för att säkerställa språkmodellernas tillförlitlighet och användbarhet i praktiska tillämpningar (Shen et al. 2023). Två av de mest framstående AI-verktygen som tillämpar stora språkmodeller och NLP teknologi idag är OpenAIs ChatGPT och GitHubs Copilot (Naveed et al. 2023; Zhang et al. 2023).

2.3.1 ChatGPT

ChatGPT definieras av Wu et al. (2023) som en intelligent chattrobot som kan leverera detaljerade svar baserade på instruktioner i en prompt. Som en del av Generativ-AI beskriver Wu et al. (2023) att ChatGPT visat sig vara ett kraftfullt AI-verktyg för en mängd uppgifter inom språkförståelse och språkgenerering, såsom flerspråkig maskinöversättning, felsökning av kod och berättelseskivning (Github, 2024) En betydelsefull skillnad från tidigare chattbotar är ChatGPTs förmågan att minnas tidigare svar i konversationen med hjälp av ML teknologi, vilket underlättar för en kontinuerlig dialog och förbättrar kvaliteten av framtida svar (Wu et al. 2023).

Genom att integrera djupinlärning, en del av ML och NLP, har användningen av generativa och förtränade transformer-arkitekturer (GPT) betydligt förbättrat funktionerna hos chattrobotar och andra AI-verktyg baserade på NLP (Shen et al. 2023). Enligt Shen et al. (2023) har ChatGPT, visat på framstående kapaciteter i att förstå och generera naturligt språk. Modellen använder transformer-arkitekturen vilket möjliggör för modellen att effektivt identifiera samband mellan olika delar av indata, som förbättrar dess förmåga att generera sammanhängande och människoliknande texter (Shen et al. 2023). Detta framsteg har direkt påverkat hur chattrobotar som GPT-3 och NLP-baserade system kan hantera mer komplexa och nyanserade dialoger, och därmed utöka deras användbarhet inom ett brett spektrum av applikationer (Shen et al. 2023).

Trots dessa nämnda styrkor har ChatGPT också sina begränsningar beskriver Shen et al. (2023), som att språkmodellen ibland kan producera trovärdiga men felaktiga svar och göra antaganden snarare än att ställa klagörande frågor vid otillräcklig information från användaren. Denna tendens till hallucinationer och att följa instruktioner utan äkta interaktion är välkända utmaningar inom NLP-modeller (Shen et al. 2023).

Framstegen inom bland annat ML och nätverksstrukturer, samt en ökad datorkraft och tillgång till mer data för träning, har lett till lanseringen av den första publika versionen av GPT-3 i slutet av 2022. Denna modell lockade över 100 miljoner aktiva användare under de första två månaderna (Wu et al. 2023). I mars 2023 lanserades GPT-4, vilket innebar en betydande uppdatering som möjliggör för användarna att inkludera bilder och dokument i sina prompts. Denna funktion låter modellen hantera ännu mer komplexa förfrågningar, såsom bildanalyser och dokumentanalyser (Wu et al., 2023).

2.3.2 *GitHub Copilot*

GitHub Copilot, som ofta benämns som en par-programmerare är en generativ-AI modell utvecklad av Github, OpenAI och Microsoft (Github, 2024). Github (2024) beskriver modellen som ett AI-verktyg som ger förslag och autokompletterar kod. Genom att analysera användarens kod i realtid kan modellen ge både förslag och autokomplettera baserat på användarens egen kod eller baserat på kommentarer i naturligt språk direkt i användarens IDE (Integrated development environment) (Github, 2024). Zhang et al. (2023) menar på att Copilot, som drivs utav OpenAI Codex, använder NLP för att förstå användarens input i naturligt språk och ML teknik för att generera lämpliga förslag till koden. Copilot, med sin kapacitet att generera kod, är ett värdefullt stöd i utvecklingsprocessen och uppges kunna ge snabba och relevanta kodförslag. Däremot belyser Zhang et al. (2023) att dess förslag ibland kan vara icke-optimala eller felaktiga och därmed kräver en högre nivå av teknisk erfarenhet på användarens sida.

Dakhel et al. (2023) utförde en jämförande studie som utforskade GitHub Copilots förmåga att generera kodlösningar jämfört med mänskliga programmerare. Studien indikerar att även om Copilot hanterar ett brett spektrum av programmeringsuppgifter, från grundläggande algoritmer till komplexa datastrukturer, varierar dess prestanda signifikant. I vissa fall genererar Copilot lösningar som är jämförbara med eller överträffar mänskliga kodstycken i effektivitet

och korrekthet, medan den i andra scenarier producerar kod som kräver ytterligare modifieringar eller felsökning (Dakhel et al. 2023; Zhang et al. 2023).

För att optimera användningen av Copilot, understryker Zhang et al. (2023) och Dakhel et al. (2023) vikten av att utvecklare besitter teknisk skicklighet samt förståelse för hur AI-baserade förslag kan integreras effektivt i mjukvaruutvecklingsprocessen. Denna insikt framhäver Copilots roll som en facilitator för en ny era inom mjukvaruutveckling, där samarbetet mellan människa och AI har potential att förbättra produktiviteten väsentligt, förutsatt att de inneboende utmaningarna hanteras med omsorg (Dakhel et al. 2023; Zhang et al. 2023).

2.4 Mjukvaruutveckling

2.4.1 Mjukvaruutveckling bakgrund

Mjukvaruutveckling omfattar skapandet, designen, testningen, distributionen, underhållet och supporten av mjukvara, vilket definieras som program som instruerar datorer att utföra specifika uppgifter (IBM, 2024). Sedan John Tukey myntade termen software år 1957, har fältet utvecklats från arbetskrävande manuella metoder där hålpunktkort betecknade instruktioner i maskinkod till avancerade högnivåspråk som liknar naturligt språk mer och mer. (Booch, 2018; Yost, 2018).

Banbrytande språk som BASIC, Cobol, Pascal och C formade mjukvaruutvecklingen under de följande decennierna (Booch, 2018; Yost, 2018). Accelerationen av utvecklingen under 1970- och 1980-talen, driven av framväxten av persondatorer som Apple II och IBM:s PC, markerade en betydande milstolpe (Yost, 2018). Dessa framsteg ledde till utvecklingen av Integrated Development Environments (IDEs), vilket revolutionerade programvaruutvecklingen (Wirth, 2008). Ökad efterfrågan och komplexitet på mjukvara under senare delen av 1900-talet drev framväxten av Objektorienterad Programmering (OOP), vilket erbjöd en mer flexibel och skalbar modell för att utveckla större system (Wirth, 2008).

Nu mera använder programmerare oftast högnivåspråk som förenklar kodning genom att spegla hur de uppfattar problem, snarare än hur maskinvaran bearbetar data (Yost, 2018). Sedan Winston Royce introducerade vattenfallsmetoden 1970, som erkände mjukvaruutvecklingens komplexitet och behovet av strukturerad organisation för att undvika misslyckanden, har flera nya metodiker utvecklats (Clarke & O'Connor, 2012).

2.4.2 AI-assisterad mjukvaruutveckling

AI-verktyg för programmering har förändrat mjukvaruutvecklingsprocessen med sin förmåga att effektivisera och automatisera testning, felsökning, analys, optimering samt generering av kod (Kuhail et al. 2024). Som tidigare nämnt har AI-verktygen bidragit till utvecklingsprocessen genom att automatisera specifik monotona moment och erbjuda förslag som syftar till att

öka utvecklarens effektivitet och produktivitet (Kuhail et al. 2024). Brynjolfsson & McAfee (2017) lyfter som tidigare nämnts också detta och menar på att AI är som mest effektivt när det används för specifika steg i en process.

Fortsättningsvis betonar Dakhel et al. (2023) att AI-baserade programmeringsverktyg, såsom GitHub Copilot, har visat sig vara särskilt fördelaktiga när de används av erfarna utvecklare för att snabba upp kodningsprocessen genom att automatiskt generera korrekt och effektiv kod för fundamentala algoritmiska problem (Dakhel et al. 2023). De noterar dock att dessa verktyg, trots deras förmåga att förbättra produktiviteten, även kan leda till problem när de används av mindre erfarna utvecklare. I synnerhet kan nybörjare ha svårigheter att identifiera och rätta till de felaktiga eller suboptimala lösningar som AI-verktyget kan generera, vilket riskerar att sänka den övergripande kodkvaliteten (Dakhel et al. 2023). Utmaningarna i tillförlitlighet och etiska övervägande som utveckling med AI-verktyg medför lyfter också Roumeliotis och Tselikas (2023) i sitt arbete och understryker behovet av noggrann övervägning och hantering när dessa teknologier implementeras i utvecklingsprocessen. Dakhel et al. (2023) utvecklar på detta och menar på att AI-verktyg bör användas med en välgrundad förståelse för dess styrkor och begränsningar, för att verkligen gynna mjukvaruutvecklingsprocessen (Dakhel et al. 2023).

2.4.3 Människa-AI samarbete

Mänskligt samarbete med AI är enligt Wang et al. (2020) inte att förlika med mänsklig interaktion med AI. Wang et al. (2020) menar på att samarbete handlar om en gemensam förståelse av mål, förebyggande uppgifts hantering och en gemensam framstegsspårning. Denna synpunkt utforskades närmare av Hamza et al. (2023), särskilt inom ramen för programmering. Hamza et al. (2023) kom fram till att AI verktyg har gått från just ett verktyg till en aktiv samarbetspartner. Deltagarna i studien använde inte bara ChatGPT för rutinmässigt arbete utan förlitade sig även på dess kreativa förmåga (Hamza et al. 2023). Den dynamiken som observerades mellan ChatGPT och programmerare indikerar att AI har blivit mer av ett samarbetsverktyg snarare än en assistent (Hamza et al. 2023). Med detta i beaktning kan man enligt Hamza et al. (2023) mer förlika AI som en samarbetsverktyg vilket Wang et al (2020) också antyder. Trots detta är det fortfarande enligt Hamza et al. (2023) fortfarande viktigt att förstå verktygens begränsningar. Exempel på detta är att ChatGPT inte kan felsöka kod i vad som refereras till som ”debugger” läge vilket leder till att den endast kan ge förslag medan programmeraren måste diagnostisera och lösa själv.

2.4.4 AI-augmentation

Intelligent augmentation (IA) beskrivs av Hassani, Silva, Unger TajMazinani & Mac Feely (2020) som en design för att stärka den mänskliga kognitionen med hjälp av AI. Detta koncept är enligt Rold, A.C (2019 citerad i Hassni et al, 2020) till för att demonstrera hur AI och människor kan komplettera varandra. IA kan vidare ses som en GPS där människan är den som kör och tar till handling medan AI rekommenderar bästa rutter fast lämnar det ultimata

beslutet till människan (Clevva, 2019 citerad i Hassani et al, 2020). Kontra AI ”ståndpunkten” som är mer benägna att tro att AI kommer att ersätta mänskliga funktioner tror IA ”lägret” på mänsklig kognition i centrum av interaktionen mellan människan och datorn (Levada, D. 2016 citerad i Hassani et al. 2020). Detta är något som Mitchell (2021) också menar på, AI har gått igenom flera cyklar av överoptimism och som sedan följts av pessimism, såkallat AI-vinter. Vilket kan tyda på en viss hög optimism från AI ”lägret” som beskrivs i Hassani et al (2020). Detta är något som uppfattas utifrån Johnson, Laurell, Ots & Sandström (2022) som beskriver att AI främst används för att förstärka människorna som jobbar inom R&D och i en ganska begränsad omfattning faktiskt automatiserar. Det finns dock skillnader mellan sektorer och bland annat mjukvaruindustrin hade en högre grad av automation enligt Johnson et al. (2022). Dock var det en liten del som bestod av automation och en större del bestående av en kombination av förstärkning och automation (Johnson et al, 2022). Detta är även hur Doraiswamy (2017 citerad i Hassani et al. 2022) beskriver den framtida AI revolutionen, det vill säga att människor lär sig använda AI som förstärkning kan bli den riktiga revolutionen. Vidare beskriver Hassani et al. (2022) att svårigheterna ligger i utbildning och att utbildare måste förstå vikten av mänsklig kreativitet i samband med AI och IA (Araya, 2019 citerad i Hassani,et al. 2022). Detta borde bli viktigt inom en snar framtid eftersom som tidigare nämnts syftade 55% av AI implementation inom R&D år 2020 att förstärka människan i arbetsprocessen (Hassani et al. 2022).

2.5 Unified Theory of Acceptance and use of Technology

2.5.1 Ramverkets delar

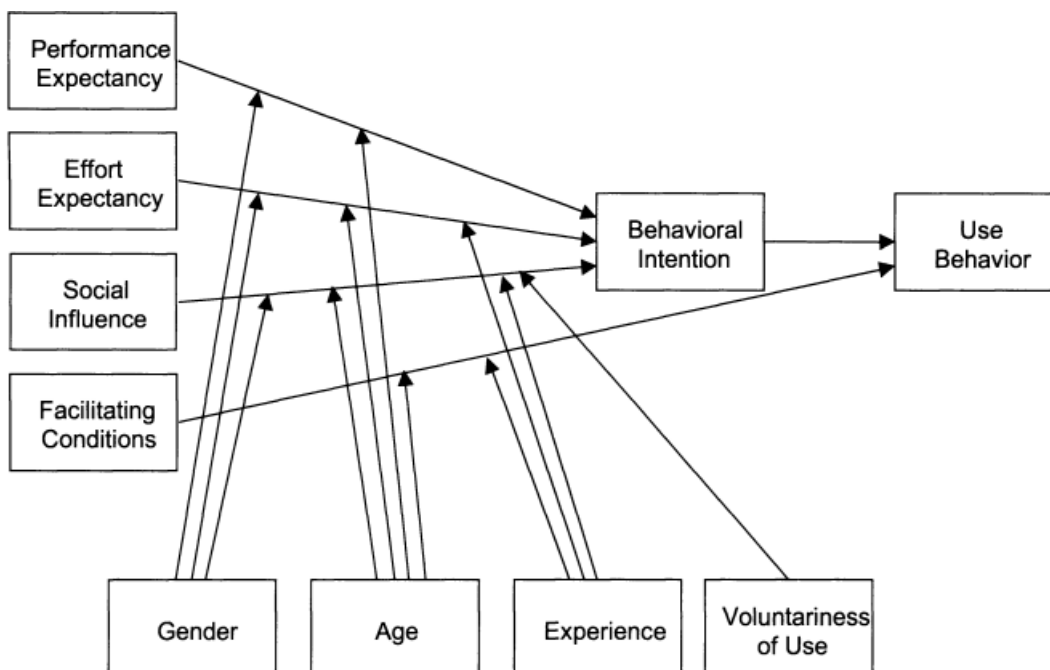
I takt med den snabba utvecklingen av generativ-AI är det av stor vikt att förstå sig på faktorer som acceptans och effektiv användning. Tidigare och även idag finns det flera modeller att välja mellan när det kommer till acceptansen av informationsteknologi (Venkatesh et al. 2003). Venkatesh et al (2003) skapade modellen Unified Theory of Acceptance and Use of Technology (UTAUT) utifrån åtta välkända modeller så som Technology Acceptance Model (TAM) och Theory of Planned Behavior (TPB). Modellen syftar åt att konsolidera och interagera de varierande insikterna från tidigare acceptans modeller genom att erbjuda ett enhetligt teoretiskt ramverk. Venkatesh et al (2003) kom fram med fyra kärnkonstrukt som bestämmande faktorer för användaracceptans och användningsbeteende. Dessa är förväntad prestanda (performance expectancy), förväntad ansträngning (effort expectancy), socialt inflytande (social influence) och underlättande förhållanden (facilitating conditions). Vidare finns det även fyra moderatörer som är ålder, erfarenhet, kön och frivillighet.

Förväntad prestanda innebär till vilken grad en individ uppfattar att användningen av en teknologi kommer bidra till arbetsprestanda (Venkatesh et al. 2003). Här spelar dock flertalet individbaserade faktorer in enligt Venkatesh et al. (2003) och han menar på att de är det högst sannolikt att effekten blir störst för yngre män kopplat till moderatörerna.

Vidare är nästa konstrukt förväntad ansträngning, detta är som det låter den förväntade svårighetsgraden associerad med att användning av teknologin. Detta konstrukt modereras av ålder, kön och erfarenhet (Venkatesh et al. 2003). Konstruktet är taget från olika delar i tidigare modeller som handlar om uppfattningen om hur lång tid det kommer ta att lära sig teknologin och hur lätt det är att använda teknologin (Venkatesh et al. 2003).

Det tredje kärnkonstruktet är socialt inflytande och det bygger på huruvida en individ uppfattar att omgivningen tycker att det är viktigt att denna ska använda den nya teknologin (Venkatesh et al. 2003). Det vill säga desto mer uppmaning människan får från omgivningen att använda en teknologi desto större är sannolikheten att individen gör det. Detta konstrukt modereras av kön, ålder, frivillighet och erfarenhet. Enligt teorin är det framför allt äldre kvinnor med mindre erfarenhet som framför allt kommer att influeras av denna aspekt (Venkatesh et al. 2003).

Det fjärde och sista konstruktet är underlättande förhållanden, detta är hur stor grad individen har förtroende för att organisationen besitter tillräckligt med infrastruktur för att kunna stödja teknologin (Venkatesh et al. 2003). Underlättande förhållande kommer att modereras av ålder och erfarenhet, det vill säga effekten kommer att bli större på äldre medarbetare som besitter mycket erfarenhet (Venkatesh et al. 2003). Denna del är också direkt kopplad till användnings beteende och inte beteendeavsikt som de andra tre konstrukten är. Det vill säga förväntad prestanda, förväntad ansträngning och socialt inflytande påverkar beteendeintentionen, underlättande förhållande i gemenskap med beteendeintentionen påverkar i sin tur användarbeteendet (Venkatesh et al. 2003).



Figur 2.2: UTAUT ramverket. (Venkatesh et al. 2003)

2.5.2 Hur UTAUT tillämpas

UTAUT har tillämpats i stor omfattning, bland annat tillämpades det av Hannah Gasch Rempel & Margaret Mellinger (2015) som använde det för att undersöka varför forskare väljer att använda vissa typer av referenshanteringsverktyg. I deras studie använde sig av kvalitativa intervjuer och observationer av arbete (Rempel & Mellinger, 2015). Rempel & Mellinger (2015) menar att samtliga av konstrukten var välpassande och pekar på att bland annat förväntad ansträngning spelade en stor roll när det kom till fortsatt användning av verktyget.

Vidare använde sig Marion Lucille Williams, Ian Peter Saunderson & Alexander Dhoest (2021) ramverket i syfte att med kvalitativ metodik ta reda på studenter uppfattning om sociala medier i akademiska bibliotek. Fynden i denna studie visar på att samtliga av konstrukten på olika grader är viktiga för studenters användning av sociala medier (Williams et al. 2021). Studien trycker bland annat på att det är väldigt viktigt med robust internet och att konstruktet underlättande förhållande således är av stor vikt för slutanvändning.

Som tidigare nämnts består UTAUT av fyra kärnkonstrukt som syftar till att bygga en förståelse om användning av teknologi (Venkatesh et al. 2003). Enligt Dwivedi et al. (2019) har UTAUT använts extensivt av forskare för att förstå sig på IT-acceptans. Blut et al. (2021) beskriver vidare att moderatorerna inte har använts extensivt inom forskning och det blir därav osäkert hur stor effektmoderatorerna har inom specifika kontexter. Dwivedi et al. (2019) beskriver också hur moderatorerna i mycket av den analyserad researchen inte används. De beskriver vidare att de kan bli relevanta endast när det finns en betydande variation i dessa moderatorer mellan individer inom samma kontext (Dwivedi et al. 2019). Inom ramen för författarnas studie kommer moderatorn erfarenhet att användas för att analysera användarnas acceptans av AI-verktyg inom programmering eftersom författarna funnit att detta varit den moderatorn med mest variation.

2.5.3 Kritik mot UTAUT och eventuella utbyggnader av modellen

UTAUT har använts extensivt inom informatikforskning och har under många år varit ett ramverk som forskare använder inom sina studier. Blut et al. (2021) utförde en extensiv metaanalys av UTAUT framework och kom fram till att både UTAUT och UTAUT 2 hade vissa begränsningar. Framför allt att ramverket enligt Blut et al. (2021) mindre robust än man tidigare trots och det finns en oro för att teorin inte replikerats tillräckligt och lämpligt. Dock finner både Williams et al. (2021) och Rempel & Mellinger (2015) insikter kring konstrukten som påverkar systemanvändning inom två ganska olika områden vilket ändå tyder på att ramverket har en viss robusthet.

2.5.4 Litteratursammanfattning

I tabellen presenteras författarna som har refererats i respektive kapitel i litteraturgenomgången.

Tabell 2.1 Litteratursammanfattning

Kapitel	Litteratur
2.1 Artificiell Intelligens 2.1.1 Definition och bakgrund 2.1.2 Maskininlärning 2.1.3 Naturlig språkbehandling	<ul style="list-style-type: none"> - Russel & Norvig (2020) - Mitchell (2021) - OECD (2024) - IBM (2024) - Mahesh (2020) - Jordan & Mitchell (2015) - Sarker (2021) - Roumeliotis & Tselikas (2023) - Zhang et al. (2023) - Agarwal & Saxena (2019) - Cambria & White (2014)
2.2 Generativ-AI 2.2.1 Adoptionen av generativ-AI i mjukvaruutveckling	<ul style="list-style-type: none"> - Nhavkar & Goel (2023) - Brynjolfsson, Li & Raymond (2023) - Nah, Zheng, Cai, Siau, Chen (2023) - Bandi, Adapa & Kuchi (2023) - Van Slyke, Johnsson & Sarabadani (2023) - Salesforce (2023) - McAfee, Rock & Brynjolfsson (2023)
2.3 AI-verktyg 2.3.1 ChatGPT 2.3.2 GitHub Copilot	<ul style="list-style-type: none"> - Kuhail et al. (2024) - Naveed et al. (2023) - Shen et al. (2023) - Wu et al. (2023) - Github (2024) - Dakhel et al. (2023)
2.4 Mjukvaruutveckling 2.4.1 Mjukvaruutveckling bakgrund 2.4.2 AI-assisterad mjukvaruutveckling 2.4.3 Människa-AI samarbete 2.4.4 AI-augmentation	<ul style="list-style-type: none"> - Booch (2018) - Yost (2018) - Wirth (2008) - Clarke & O'Connor (2012) - Brynjolfsson & McAfee (2017) - Wang et al. (2020) - Hamza, M. et al. (2023) - Hassani et al. (2020) - Johnson et al. (2022)
2.5 Unified Theory of Acceptance and use of Technology 2.5.1 Ramverkets delar 2.5.2 Hur UTAUT tillämpas 2.5.3 Kritik mot UTAUT och eventuella utbyggnader av modellen	<ul style="list-style-type: none"> - Venkatesh et al. (2003) - Hannah et al. (2015) - Williams et al. (2021) - Dwivedi et al. (2017) - Blut et al. (2021) - Rempel & Mellinger (2015)

3 Metod

I metodkapitlet redogör författarna för sin insamlingsmetod och litteratursökning. Motivationen till metodvalet beskrivs, vilket inkluderar en beskrivning av semistrukturerade intervjuer, utformande av intervjuguiden, etiska övervägande, urval av respondenter och valet av AI-assistera transkribering. Det redogörs också för kategoriseringen av data, grunden till studiens analys, och slutligen utvärderas validiteten och reliabiliteten av datan.

3.1 Metodval

För studien har en kvalitativ metod valts som undersökningsmetod. Syftet med studien är att undersöka vad utvecklarens erfarenhet och acceptans är kring användandet av AI-verktygen för mjukvaruutveckling. Studien har genomfört fem intervjuer från respondenter som alla har praktisk erfarenhet av AI-verktygen såsom ChatGPT och Github Copilot i programmerings-syfte. Semi-strukturerade intervjuer genomfördes för att få en inblick i utvecklarens erfarenhet och acceptans av användandet.

3.2 Forskningsfilosofi, Kvalitativ metod

Enligt Oates et al. (2022) är en kvalitativ metod en datainsamling av icke numeriska data såsom ord, ljud, bilder genom till exempel inspelade intervjuer, dokument och hemsidor. Fortsättningsvis stödjer Jacobsen (2002) detta och beskriver kvalitativa metoder som insamling av data i form av ord vilket öppnar upp för att se fler nyanser, gå på djupet och öppnar upp för att se kontextuella aspekter. För att verkligen förstå sociala fenomen bör man undersöka hur människor tolkar den sociala verkligheten. Detta kan enligt Jacobsen (2002) inte göras på annat sätt än att observera människor, se vad de gör och säger samt låta dem tala med egna ord. För att få insikt i utvecklarens erfarenheter med AI-verktyg i programmering har kvalitativa semistrukturerade intervjustudie genomförts som följer en interpretivistisk ansats. Den interpretivistiska ansatsen möjliggör för att undersöka individens subjektiva upplevelser, vilket instämmer med studiens mål att undersöka utvecklarens erfarenhet och acceptans till AI-verktyg.

3.2.1 Kritik mot metoden

Trots den så kallade kvalitativa vändningen där kvalitativa metoder dominerat samhällsvetenskaplig forskning de senaste decennierna har kvalitativa metoder kritiserats för att inte förse undersökaren med tillräckligt med information om hur analysen av kvalitativa data bör ske (Oates et al. 2022). Eftersom kvalitativa metoder ofta utgår ifrån en datainsamling av text i form av till exempel intervjuer finns det en risk att undersökaren blir överväldigad av volymen data som kan genereras av relativt få intervjuer (Oates et al. 2022). Metoden kritiseras för detta då undersökaren löper risk att förlora förmågan att identifiera mönster och teman i datan

(Oates et al. 2022). Författarna har valt att genomföra fem semistrukturerade intervjuer med en snitttid på 28 minuter för att minimera risken att känna sig överväldigade av volymen samt valt att använda sig av automatiska transkriberingsverktyg som underlättar transkriberingsprocessen.

Fortsättningsvis så kritiseras kvalitativa metoder för att vara subjektiv och tenderar att baseras på undersökarens intryck snarare är fakta (Oates et al. 2022). Oates et al. (2022) menar på att tolkningen av data sker närmare undersökaren, på så sätt kan deras identitet, antagande och bakgrund leda till preliminära slutsatser. Genom att använda semistrukturerade intervjuer har författarna skapat en intervjuguide, vilket agerar som en ram för samtalet där frågor relevanta för forskningsfrågan ställts, samtidigt som respondenternas upplevelser och åsikter ges utrymme att uttryckas fritt. Respondenternas svar har även triangulerats, där flera respondenters svar används för att bekräfta eller komplettera varandra.

Bell, Bryman & Harley (2019) kritiserar också kvalitativa metoder för att ha en för begränsad omfattning. Dem menar på att observationer av ett mindre antal deltagare inom till exempel en organisation inte kan generaliseras till andra omgivningar. Även om studien har behandlat ett mindre antal deltagare inom en organisation har semi strukturerade intervjuer valts för att försöka få en djupgående förståelse för respondenternas acceptans och erfarenhet.

3.2.2 Valet av intervjuer som insamlingsmetod

En intervju är enligt Oates et al. (2022) en specifik konversation mellan individer med vanligtvis en del antaganden (oftast osagda) som inte finns i vanliga konversationer. Vidare berättar Oates et al. (2022) att en av personerna ska ha en anledning att genomföra intervjun, det vill säga att det är en strukturerad konversation med syftet att personen som efterforskar ska få ut information från de andra parterna. Intervjuerna i studien har genomförts av två undersökare och en respondent som blir undersökt åt gången. Vidare så har intervjuerna haft som mål att genomföra en ostrukturerad konversation med syftet att få information kring respondenternas erfarenhet och acceptans av AI-verktyg i programmeringssyfte.

Jacobsen (2002) beskriver vidare vad som bör tas i beaktning när man väljer hur slutet eller öppen intervjun ska vara. Två motargument kan enligt Jacobsen (2002) anföras mot en helt öppen intervju, Det första är att en ansats utan strukturering leder till att det blir extremt resurskrävande och näst intill omöjligt att analysera. Det andra argumentet mot en helt öppen intervju är att det alltid tillkommer en viss form av strukturering omedvetet, även om det är omedvetet tar man med sig fördomar innan man börjar samla in data. Oates et al. (2022) definierar precis som Jacobsen (2002) att det finns grader av strukturer i intervjusammanhang. Dessa är Strukturerade intervjuer, Semi-strukturerade intervjuer och ostrukturerade intervjuer. Det som Jacobsen (2002) förespråkar är det som Oates (2022) definierar som semi-strukturerad intervju.

Studien har som fokus att ta reda på acceptansen och erfarenheten som utvecklare haft med användningen av AI-verktyg i programmeringssyfte, detta ledde till att beslut togs om att

intervjuerna ska vara semi-strukturerade. En semi-strukturerade intervju beskrivs av Oates et al (2022) som en intervju där man har fördefinierade teman som täckas och frågor som ska ställas. Vidare beskriver Oates et al (2022) att man kan ställa fler frågor än vad som förberetts för och intervjupersonen kan gå djupare på frågorna som ställs och lyfta problematik utanför de initiala frågorna. Detta är välpassande för studien då studien avser att gå mer på djupet om hur individuella utvecklare ställer sig mot AI-verktyg.

3.3 Datainsamling

3.3.1 Litteraturkapitel

En litteraturgenomgång genomfördes för att samla in en övergripande förståelse av den nuvarande vetenskapliga litteraturen samt hitta relevanta källor till studien. Detta kan enligt Jacobson (2018) bidra till en bättre förståelse för ämnet och bidra till att identifiera kunskapsluckor. De källor som författarna av studien har valt att använda består av tidigare forskning, böcker, studier, organisatoriska webbkällor och journaler publicerade av kända utgivare. Processen involverade att söka efter relevant litteratur, främst via Google Scholar, LUBsearch, Springer-Link och Science Direct vilket är kvalitetsgranskade. Ett flertal källor från kända utgivare hämtades direkt från kursmaterial från systemvetenskapliga kandidatprogrammets kurs INFA40. Metodböckerna som används har lånats från LUSEMS:s bibliotek.

För att få en övergripande förståelse gick litteraturgenomgången igenom begrepp som Artificiell Intelligens (AI), Maskininlärningsalgoritmer, Generativ-AI, Natural Language Processing (NLP), Large Language Models (LLM), ChatGPT, Github Copilot, Mjukvaruutvecklingsprocessen, Human AI collaboration, AI-augmentation och AI-assisterad programmering. De huvudsakliga nyckelord som användes för att söka efter relevant litteratur online är listade i **tabell 3.1**.

Studiens författare har valt relevant litteratur utifrån tre kriterier främst:

Det första kriterium som beaktades vid val av relevant litteratur var dess aktualitet. Litteraturens aktualitet beaktades extra noga eftersom ämnet som undersöks är relativt nytt och utvecklas i snabb takt. Många av områdena som studien behandlar som till exempel LLM, NLP och Maskininlärning är tekniker som har haft stora genombrott i närtid och är därför viktiga att ha litteratur som är aktuell. Eftersom aktualitet har varit största prioritet vid litteratursökningen finns det källor som inte har lika högt antal citeringar som de äldre källorna. Det existerar också en andel ”grå litteratur” vilket är litteratur från webbsidor, tidskrifter och rapporter (Karolinska Institutet Universitetsbiblioteket, 2023). För de områden i uppsatsen som inte haft en lika snabb utveckling i närtid har aktualitet inte beaktats lika mycket och antal citeringar och vetenskapliga publiceringar har i stället prioriterats. För dessa områden har en historisk kontext beaktats och äldre välgrundade artiklar och böcker valts.

Det andra kriteriet som beaktades vid valet av litteratur var ifall källan var hade blivit referensgranskad i en vetenskaplig publicering. Författarna valde att använda sig av vetenskapliga publiceringar för att öka trovärdigheten, detta på grund av att publiceringar blivit granskade av andra experter inom områdena som berörs (Rienecker, L. Stray Jörgensen, P. & Hedelund, L. 2014). Författarna hade som mål att små många artiklarna som möjligt skulle vara kvalitetsgranskade vetenskapliga publiceringar.

Det tredje kriteriet som beaktades vid insamling av litteratur var antalet citeringar som källan hade vid insamlingstiden. Rienecker et al. (2014) menar på att antalet citeringar som ett verk har talar för hur inflytelserikt verket har varit inom området. Därför har författarna haft som mål att inkludera källor med ett högt antal citeringar. Detta för att ytterligare säkerställa att källorna är trovärdiga. I vissa fall har litteraturens aktualitet prioriterats framför antal citeringar då aktualitet ansågs viktigare än antal citeringar på grund av områdets snabba utveckling och författarnas intentioner att inkludera litteratur som är så nära i tid som möjligt. När ”grå litteratur” användes kunde inte antal citeringar inte beaktas då hemsidor inte visar citeringar likt Google Scholar och LUBsearch gör.

3.1 Litteratursökningstabell: Tabell över teman med respektive nyckelord som användes vid litteratursökningen.

Nyckelord:	Kombinerat med:
- Artificial Intelligence	- “and”
- AI-pair programming	- “risks”
- ChatGPT	- “benefits”
- GitHub Copilot	- “efficiency”
- UTAUT	- “productivity”
- AI-augmentation	- “effect”
- LLM	
- NLP	
- Software Development Process	

Venkatesh et al (2003) var vår äldsta källa som användes i litteratursökningen och som grund för vår analys med hjälp utav ramverket UTAUT. Viswanath Venkatesh anses allmänt som en av de mest inflytelserika forskarna inom området informationssystem, både vad gäller främsta tidskriftspublikationer och citat. Källan kan därmed anses trovärdig och användas.

3.3.2 Utformande & Utförande av Intervju

Oates et al. (2022) beskriver att individuella intervjuer behöver planering och kunskap för att vara lyckade. Det är också enligt Oates et al. (2022) viktigt att intervjuaren är punktlig, professionell, trevlig, mottaglig och neutral, eftersom intervjupersonens svar i viss utsträckning beror på hur intervjuaren uppfattas av intervjupersonen. Detta lyfts även av Jacobsen (2002) som hävdar att i intervjuer ansikte mot ansikte så spelar intervjuarens uppträdande en stor roll.

Intervjuerna bör enligt Oates et al. (2022) planeras och frågor som ska lyftas bör vara formulerade. Intervjun bör vidare enligt Jacobsen (2002) inleda intervjun med att kort berätta om undersökningen. Det som enligt Jacobsen (2002) är viktigt att få med är:

- Vem vi som intervjuare är
- Vad som är ändamålet med undersökningen
- Hur informationen som framkommer i intervjun ska användas
- Vem som har beslutat att undersökningen ska genomföras

Man ska också enligt Jacobsen (2002) om respondenterna undrar någonting i samband med intervjun kunna svara på det. Med bakgrund till detta och i enlighet med Jacobsen (2002) lyfts därav en intervjuguide med tema och fast ordningsföljd som

Intervjuerna genomfördes i person och via Zoom meetings. Valet gjordes baserat på vad respondenterna föredrog och som var möjligt utifrån respondenternas schema. De semi-strukturerade intervjuerna utgick utifrån en intervjuguide som har fördefinierade teman som täckas och frågor som ska ställas. Detta går i enlighet med vad Oates et al. (2022) beskriver som en semi-strukturerad intervju vilket även lät oss ställa fler frågor än vad som förberetts för och intervjupersonen kan gå djupare på frågorna som ställs och lyfta problematik utanför de initiala frågorna. Intervjuguidens tema utgår ifrån de fyra konstrukt i ramverket UTAUT och syftar till att undersöka acceptansen och erfarenheten. Intervjuerna inkluderade även en introduktions del som syftar till att introducera respondenten till studien och dess syfte samt samla information om deras bakgrund och erfarenhet till ämnet. Avslutningsvis inkluderades även frågor där respondenterna fick lägga till saker som ansetts ha missats och tankar kring framtiden för AI-verktyg i programmering. Detta fanns inte med i UTAUT modellen men var av intresse för undersökarna. Undersökningsmodellen som författarna använde vid utformningen av intervjufrågorna med respektive tema och kapitel hittas i Appendix (F).

3.3.3 Motivering av Intervjuguide

Utifrån de områden och ramverk som valts att användas i litteraturgenomgången så har en mall i form av en intervjuguide formats för att standardisera samtliga intervjuer. Detta säkerställer att alla teman som avses gå igenom tas upp menar Jacobsen (2017). Intervjuguiden är utformad efter sex teman som avser att täcka UTAUT:s fyra kärnkonstrukt, för att få en förståelse över respondenternas acceptans av användningen av AI-verktyg inom mjukvaruutveckling och en av ramverkets moderatorer, Erfarenhet, en av UTAUT:s moderatorer som avser att styrkan i sambanden mellan förväntan om ansträngning, socialt inflytande och underlättande förhållanden (Venkatesh et al. 2023). Efter litteraturgenomgången lades ett sjätte tema till, Framtid, som avser att undersöka respondenternas tankar kring framtiden av AI-verktyg inom programmering. Introduktion och avslutningsfrågor har även inkluderats för att introducera ämnet för respondenten samt runda av intervjun där respondenten får chans att ta upp saker som hen anses missats.

Intervjuerna som genomfördes var semi-strukturerade vilket lät intervjuarna ställa frågor utöver de som fanns i intervjuguiden och lät respondenterna utveckla sina svar fritt. Detta gjordes för att öka kvaliteten på intervjun och för att möjliggöra följdfrågor på intressanta svar. Detta sågs kunna leda till insikter och vinklar på ämnet som inte tagits upp av intervjuarna. På grund av den semi-strukturerade naturen av intervjun tog inte alla frågor upp men alla teman togs upp. Intervjuerna genomfördes efter att större delen av litteraturkapitlet skrivits och varierade mellan 25 och 35 minuter där merparten genomfördes i person, med två som fick genomföras via Zoom på respondenternas begäran.

3.3.4 Urval av Respondenter

Den första kontakten med potentiella respondenter för studien skedde genom e-post eller verbalt, där syftet för studien presenterades. Vidare så förklarades även upplägget av intervjuprocessen och de potentiella respondenterna fick chans att ställa frågor om studien. Om respondenten verkade positiv till deltagandet hölls ytterligare kontakt, antingen via e-post eller i person. Respondenterna som tillfrågades valdes utifrån deras erfarenheter och koppling till ämnet som studien undersöker. Urvalet av respondenter var inte helt och hållet optimalt då författarna initialt fick många nej till deltagande och tidsramen för arbetet inte lät oss utforska alla möjliga respondenter som vi kanske hade önskat. Initialt så avsåg författarna att intervjua programmerare som arbetade med mjukvaruutveckling skarpt då detta ansågs optimalt för att undersöka acceptansen och erfarenheten. Studiens valda respondenter bestod av tre lärarassistenter (labbhandledare) vid Lunds universitets Institution för Informatik, En lärare för samma institution samt en student från Lunds Tekniska Högskola Datasektion. Alla respondenter valdes på grund av deras egen erfarenhet med AI-assisterad programmering.

3.2 Intervjuobjekt: Respondenter med respektive organisation, intervjutyp, yrkesroll och tid.

Respondent	Yrkesroll	Organisation	Intervjutyp	Tid	Datum
R1	Labbhandledare	Institutionen för informatik	Semi-strukturerad Zoom	27 min	2024-04-19
R2	Labbhandledare	Institutionen för informatik	Semi-strukturerad Personlig	26 min	2024-04-20
R3	Lärare	Institutionen för informatik	Semi-strukturerad Personlig	36 min	2024-04-22
R4	Student	Datasektionen LTH	Semi-strukturerad Personlig	25 min	2024-04-23
R5	Labbhandledare	Institutionen för informatik	Semi-strukturerad Zoom	25 min	2024-04-28

3.3.5 Transkribering

I undersökningen kommer den kvalitativa metoden i form av semi-strukturerade intervjuer transkriberas med hjälp av mjukvaran WhisperAI. En Automatic Speech Recognition (ASR) modell byggd av OpenAI, som assisterar med transkribering av inspelade ljudfiler (OpenAI, 2022). Modellen har tränats på över 680,000 timmar av flerspråkiga data från internet och besitter enligt OpenAI (2022) en robusthet och noggrannhet som närmare sig mänsklig nivå. Oates et al. (2022) hävdar att undersökaren bör räkna med omkring fem timmar transkribering för varje timme av intervju som genomförs, mer ifall undersökaren långsammare på att skriva. Transkriberingshastigheten hos WhisperAI beror på ett flertal faktorer, Whisper är en paraply-term för ett flertal modeller av olika storlek som erbjuder bättre noggrannhet i utbyte mot lägre transkriberingshastighet (OpenAI, 2022). Även kvaliteten av ljudfilen påverkar hastigheten (OpenAI, 2022). WhisperAI-modellerna är inte bara avsedda för taligenkänning utan också för andra uppgifter som talöversättning och tal till text på flera språk (OpenAI, 2022). WhisperAI:s laddades ner lokalt på författarnas dator under studien där samtliga inspelade ljudfiler från respondenterna transkriberades. Transkriberingarna tog mellan två och tre timmar var och analyserades även manuellt av författarna för att eliminera potentiella fel som AI modellen hade gjort.

3.4 Analys

3.4.1 Förberedande analys

Analysprocessen i studien består av en förberedande analys och en innehållsanalys av materialet. Innan huvudanalysen kunde genomföras samlades materialet in och sammanställdes för att underlätta för huvudanalysen. Ljudfilerna sparades lokalt och transkriberades automatiskt med WhisperAI. Textfilen som genererades gick igenom av båda författarna och jämfördes med ljudfilen för potentiella fel vid transkriberingen.

3.4.2 Innehållsanalys

Bryman, Bell och Harley (2019) menar på att innehållsanalys med hjälp utav kodning är vanligt förekommande inom kvalitativ forskning, där innehållsanalys syftar till att hitta och identifiera det underliggande temat i datan (Bryman, Bell, Harley, 2019). Transkriberingarna från intervjuerna har kategoriserats utifrån UTAUT:s fyra kärnkonstrukt vilket utgör grunden för analysen (Venkatesh et al, 2003), med tilläggen erfarenhet och framtid. För att tydliggöra identifieringen av kategorierna i transkriberingen har de olika kategorierna blivit tilldelade en färg och en bokstav eller bokstavskombination. Under transkribering har data som av författarna anses tillhöra en kategori färglagts med tillhörande färg och bokstäver. Avslutning och

introduktions kategorin i intervjuguiden har valts att inte tilldelas en färg då det syftar till att introducera intervjuobjektet till intervjun samt runda av intervjun. Svar som kan tolkas som passande till flera av delarna har färgkodats med den färg som passar huvudsakligen, men tilldelats de bokstäver sammanhörande de överlappande ämnena.

3.3 Kodningstabell: UTAUT konstrukt med respektive färg och bokstavkod

Kategori	Färgkod/Bokstavkod
Förväntad prestation	Gul / FP
Förväntad ansträngning	Röd / FA
Social Påverkan	Blå / SP
Underlättande medel	Grön / UM
Erfarenhet	Rosa / E
Framtid	Brun / F

3.5 Etik

3.5.1 Etiska övervägande

Enligt Oates et al. (2022) finns det fyra etiska övervägande som blir viktiga att beakta för författarnas studie. Dessa är rättigheten dra sig ur studien, rättigheten att ge ett informerat medgivande, rättigheten att vara anonym och rättigheten för sekretess. Rätten att dra sig ur helt eller delvis ska intervjupersonen alltid ha oavsett om det försämrar delar av uppsatsen (Oates et al. 2022). Detta måste sägas tydligt till intervjupersonen (Oates et al. 2022). Vidare ska intervjupersonen enligt Oates et al. (2022) ha rätten att ge ett informerat medgivande, det vill säga de ska vara helt medvetna av vad efterforskningen innebär och förstå då kan de ge sitt fullständiga medgivande. Personer direkt involverade i forskning ska också enligt Oates et al. (2022) ha rätten till anonymitet, om personen skulle vilja skydda sin identitet och plats så får man ge dem pseudonymer i forskningen istället.

För att uppnå en hög etik har ovannämnda principer tagits i beaktning vid genomförandet av studien. Speciellt vid direkt involverade i studien har det varit av stor vikt att följa övervägandena som Oates et al. (2022) lyfter, det vill säga att respondenterna ska bli informerade om samtliga av rättigheterna som nämnts. Vidare har studien en tydlig design för att säkerställa tillförlitlighet och har utformats på ett rättvist och objektiva sätt, det vill säga alla deltagare i studien har behandlats på samma vis och med ärlighet i åtanke.

Under genomförandet av studiens semi-strukturerade intervjuer har Ekonomihögskolans riktlinjer för behandling av personuppgifter i studentarbeten följts. Riktlinjerna behandlar grundläggande principer som ska uppfyllas vid hantering av personuppgifter och menar på att informerat samtycke från respondenten måste finnas. Detta stöds av vad Bryman, Bell, Harley (2019) påstår, de menar på att samtycke innebär att intervjuobjekten bör ha tillräckligt med information om studiens syfte, intervjuprocessen, och hur deras svar kommer användas (Bryman, Bell, Harley, 2019). Därför har ett samtyckes formulär tillhandahållits av författarna till intervjuobjekten innehållande information om studiens syfte, hur resultatet kommer lagras och användas och information om deras sekretess. Tillsammans med ett verbalt samtycke under den inspelade intervjun anser vi att intervjuprocessen går i enlighet med riktlinjerna från Institutionens och Bryman, Bell, Harley (2019). Som tidigare tagits upp i studien så används AI-verktyget WhisperAI för transkribering, WhisperAI kan laddas ner och sparas lokalt i stället för i ett moln vilket gör det till ett lämpligt verktyg för automatiserad transkribering. Detta eftersom man kan med säkerhet se till att data inte skickas till en tredje part. Vidare i detta övervägande har studien tagit hänsyn till LUSEM:s riktlinjer för hantering av persondata (Lund University School of Economics and Management, 2024).

3.6 Datakvalitet

Reliabilitet och validitet av insamlade undersökningsdata är två viktiga kriterier för att säkerställa och bedöma kvaliteten av forskningen (Bell, Bryman och Harley, 2019). Däremot nämner Bell, Bryman & Harley (2019) att termen validitet kan antyda mätningar av någon sort, vilket klassas mer som kvantitativa metoder. Det är därför av stor vikt att bedöma ifall validitet är ett lämpligt kriterium för att bedöma kvaliteten (Bell, Bryman och Harley, 2019).

3.6.1 Validitet & reliabilitet

För att kunna bedöma kvaliteten av undersökningen menar Jacobsen (2017) att validitet och reliabilitet är två viktiga faktorer. Enligt Jacobsen (2017) hjälper dessa kriterier undersökarna att granska kvalitativa insamlingsmetoder och därmed bedöma giltigheten av de dragna slutsatserna. Bell, Bryman och Harley (2019) föreslår att undersökaren kan assimilera kriterierna validitet och reliabilitet i kvalitativa metoder utan att förlora dess betydelse bortsett från framträdandet av mätning. Till exempel argumenteras det att validitet, reliabilitet och generaliserbarhet är tre olika sätt att mäta datans övertygande styrka, kvalitet och potential genom användningen kvalitativa metoder, principer och konventioner (Bell, Bryman och Harley, 2019). Detta tyder på att kriteriet validitet refererar till att det som uttrycks, identifieras, observeras eller mäts ska stämma överens med det som faktiskt utförs (Bell, Bryman & Harley, 2019).

Reliabilitet kan delas in i extern reliabilitet och intern reliabilitet, där extern reliabilitet talar om till vilken grad en undersökning kan återskapas (Bell, Bryman & Harley, 2019). De vill säga, hade resultatet blivit det samma om man utförde studien på nytt. Det är ett svårt kriterium att nå med kvalitativa metoder då den sociala miljö och förutsättningarna som observeras

inte kan stanna oförändrad (Bell, Bryman & Harley, 2019). Om studien hade replikerats finns det en möjlighet att respondenterna hade gett ett annat, beroende på faktorer som när intervjun tog plats på dagen eller ifall svaren baserats på slump och tillfällighet. Bryman, Bell, Harley (2019) lyfter också att forskarna kan tolka samma svar på olika sätt beroende på deras uppfattning. Intervjuguiden har därför utformats på ett sätt som ska ge respondenterna möjlighet att ge ett svar som inte påverkas av tillfällighet. I avslutningen har till exempel respondenterna fått möjlighet att lyfta egna tankar kring ämnet som de anses missats under intervjun.

Intern reliabilitet referera till faktumet att flera undersökare i samma studie kan ha olika åsikter om vad de observerar i undersökningen (Bell, Bryman & Harley, 2019). Denna studie har utförts utav två undersökare där alla intervjuer har genomförts tillsammans, undersökningsmodellen har tagits fram genom diskussion och analysen av transkribering gjorts gemensamt för att undvika en låg intern reliabilitet.

Likt reliabilitet kan även validitet delas in i intern och extern. Intern validitet menar författaren syftar till ifall undersökarnas observationer och teoretiska idéer matchar varandra och är viktigt för att undersökningen ska nå stark intern validitet (Bell, Bryman & Harley, 2019). Både intern reliabilitet och intern validitet är därmed viktiga aspekt att ta i åtanke när undersökningen utförs av flera individer. Kritiken angående generaliseringsförmågan av kvalitativa metoder som Bell, Bryman & Harley (2019) lyfter grundar sig i extern validitet. Extern validitet syftar till i vilken grad insamlingsdatan kan generaliseras till andra sociala miljöer, speciellt med kvalitativa metoders relativt små omfattningar (Bell, Bryman & Harley, 2019). Eftersom denna studies respondenter endast berör två institutioner är validiteten inte särskilt stark på en organisatorisk nivå. Däremot stärks validiteten på en individuell nivå eftersom studiens respondenter anses kunniga inom ämnet, har mycket erfarenhet av verktygen, sett till hur länge de funnits.

4 Empiri

4.1 Erfarenhet

4.1.1 Användning av AI inom programmering

Något som är tydligt är att alla respondenter anger att de använder AI-verktyg för ett programmeringssyfte i stor utsträckning, R1, R2, R3 och R5 använder ChatGPT och Github Copilot medan R4 använder ChatGPT och Gemini. När tillfrågande om när hur länge respondenterna använt AI-verktyg angav samtliga att de började använda ChatGPT strax efter lansering. Copilot som R1, R2, R3 använder började de använda strax efter lansering med, R5 som använt Copilot berättar dock att hen inte använder det i stor utsträckning. Vidare berättar respondenterna att de använder AI-verktygen primärt för att lära sig inom ämnet, generera och felsöka kod och menar på att införandet har effektiviserat alla dessa processer. R2 berättar att hen föredrar att programmera med AI-verktyg och beskriver att det hjälper väldigt mycket i vissa uppgifter, detta stöds av R5 som också berättar att hen föredrar att programmera med stöd av AI-verktyg. När respondenterna blev tillfrågade om vad det primärt använder AI-verktyg för i programmering så berättar R1 att hen använder Copilot för att snabbare kunna skriva kod, specifikt kod som är ”boilerplate” medan ChatGPT primärt används för felsökning och förklaring av existerande kod. Detta stämmer överens med R2 som använder det för lätta uppgifter och felsökning och R5 som berättar att hen främst använder AI-verktyg för att snabbt skriva kod som är standardiserad. Samtliga respondenter är eniga om att införandet av dessa AI-verktyg har haft en positiv effekt på programmering i sin helhet.

Trots att R1, R2 & R3 berättar att det har sett en ökning i produktivitet och vittnar om fördelar som till exempel stöd för problemlösning, tidsbesparing och underlättande av lärande och förståelse. Så är alla respondenterna eniga om att användandet av AI-verktyg inom programmering kräver en förståelse och granskning av förslagen som AI-verktygen producerar för att undvika fel och att AI-verktyget jobbar emot en.

R1 som har använt båda verktygen sedan lansering exemplifierar att införandet av AI-verktyg i processen har gett en positiv inverkan och dess monumentala inflytande;

” [...] Särskilt Copilot skulle jag säga att det går snabbare. Man kanske kan jämföra det med att skriva på papper jämfört med att skriva i Word (Appendix A #16)

Fortsättningsvis menar R1 att hen föredrar att programmera med AI-verktyg, främst eftersom det uppges ökar effektiviteten, produktiviteten och underlättar felsökningsprocessen. Detta har heller inte komprimerat kodkvaliteten utan snarare förbättrat den uppges R1

” [...] När jag har förstått vad den ger mig så skulle jag inte säga att kodkvaliteten inte blivit sämre, snarare bättre (Appendix A #18)

R2 som också använt båda AI-verktygen i programmeringssyfte sedan lansering antyder att införandet av verktygen har varit en bra sak som likt R1 har ökat effektiviteten vid kodning och felsöknings. Både R1 & R2 lyfter att de har haft erfarenhet av AI-hallucination eller att verktygen producerat felaktigt resultat och vilselett.

Något som R2 exemplifierar:

” [...] Det hjälper jättemycket på vissa uppgifter, men det kan också vilseleda lite. (Appendix B #22)

R3, R1, R4 & R5 belyser alla erfarenheter av felaktiga eller missvisande kodförslag under användandet av AI-verktyg. R1 beskriver hur hen fastnat i arbetet på grund av AI-hallucinationer samtidigt som R2 & R4 vittnar om vilseledande och felaktig information. Trots den upplevda effekten av AI-verktygens fallgropar så menar alla respondenterna att de fortsatt använder verktygen i programmeringssyfte då det upplevda fördelarna väger ut nackdelarna. Respondenterna delar åsikten att AI-verktygen inte är perfekta och kan producera felaktigt material men att det handlar om att använda verktygen på korrekt sätt. R4 påpekar detta och berättar att när verktygen ger en ett felaktigt svar så kan det bero på att man inte ställt frågan på korrekt sätt. Något som R5 också berättar om, när hen säger att man kan behöva bråka lite med den och att man ibland får output som är mer komplicerad än den behöver vara.

R3 exemplifierar detta:

” [...] . Om du inte är medveten då och bara kör på allting den ger dig så kan det bli problem, så kunskap är fortfarande viktigt (Appendix C #58).

4.2 Acceptans

4.2.1 Förväntad prestanda

När det kommer till att använda AI-verktyg inom programmering så rådde det konsensus om att arbetet går snabbare med assistansen. ChatGPT verkar oftast användas för att lära sig förstå nya saker, söka efter svar på problem och göra checkar inom specifika kod stycken. Medan Github Copilot används som ett verktyg för att effektivisera programmering genom att snabbt generera stora mängder kod.

R1 talar om hur hen väljer att använda Copilot för att för programmering.

” [...] Särskilt för klasser med metoder som är väldigt standard. Boilerplate kan man väl kalla det” (Appendix A, #14).

R1 påstår vidare att hen tycker att det är bra att programmera med Copilot, sen när det kommer till inlärnings inom exempelvis ett nytt kodspråk så vill hen inte använda det i syfte att lära sig språket bättre. ChatGPT använder respondenten dock framför allt i felsöknings syfte, Vidare påstår respondenten att ChatGPT är behändigt för att förstå kod. AI-verktyg för R1 fungerar som ett effektivitetsverktyg, inlärningshjälp och felsökningsverktyg. R5 använder ChatGPT på ett liknande vis som R1 använder Github Copilot, genom att använda det för att generera ganska standardiserad kod.

Felsökningsaspekten är något som R2 också syftar mot som säger att det hjälper väldigt mycket inom vissa uppgifter men att det också kan vilseleda och generera fel output. R2 använder AI inom programmering ganska snarlikt R1, hen tycker att det går snabbare att använda sig av AI för felhantering och när man ska göra relativt lätta uppgifter. Detta i syfte att människan kan tänka mer på de svåra problemen så att man inte fastnar på små kodproblem. Dock lyfter R2 också att AI är ganska dålig på det den är mindre tränad på och att den generellt effektiviserar men att AI-hallucinationer och dylika problem kan stötas på.

R2 exemplifierar upplevelser av AI-verktygs negativa aspekter

” [...] den kan ju tolka min fråga helt fel och generera fel output ” (Appendix B, #24)

R3 menar också att AI-verktyg höjer effektiviteten, utifrån perspektivet på Github Copilot så kan den generera extremt mycket kod på kort tid för att få själva programmeringen att gå fortare. Vidare säger R3 att hen har använt ChatGPT och Github Copilot för att snabbt kunna skaffa en förståelse för nya ramverk och för att kunna skapa förevisningsexempel. För att göra kurser mer användarvänliga mot MAC användare så gjorde hen en ganska omfattande omställning inom programmeringskurser med hjälp av AI.

Citat från R3 som exemplifierar AI-verktygs förmåga att effektivisera arbete:

” [...] För att kunna göra den omställningen, och jag gjorde den på rätt kort tid. Det hade jag aldrig kunnat göra utan Copilot och ChatGPT. Det hade jag aldrig kunnat ” (Appendix C, #12)

R4 har använt sig av AI inom ett ganska olikt programmeringsområde kontra de andra respondenterna. Hen använder enbart ChatGPT men säger att det är effektivt att använda det när det är något nytt som man ska ta sig an, exempelvis när man ska ta sig an ett nytt bibliotek. R4 påstår, liksom R1 & R2, att AI-verktyg är bra för att felsöka inom olika områden. Men när det kommer till objektorienterad programmering påstår R4 att det minskar tiden det tar att skriva koden.

R5 använder sig också av AI när hen programmerar och säger att hen föredrar att programmera med AI-assistans. Respondenten säger vidare att hen främst använder det för att göra standardiserade kod uppgifter så att hen kan jobba mer med den kreativa aspekten. Men att det också beror på vad man går in med för förutsättningar. Men liksom många av de andra respondenterna använder respondenten R5 ChatGPT i pedagogiskt syfte för att lära sig mer om

programmering. Dock säger respondenten också att det är bra att inte använda sig av verktygen för mycket för att kunna lära sig programmera bättre.

Vid frågan om respondenten upplever om hen kan angripa snabbare problem avslutade hen med att betona sin upplevelse av AI assisterad programmering.

” [...] Men jag tycker ändå att det nog har blivit bättre på den bemärkelsen, att jag känner att jag kan göra mer med AI än vad jag kan utan ” (Appendix E, #38)

Sammanfattningsvis använder respondenterna AI-verktygen och ser det som en effektiviseringsfaktor inom programmering även om användningsområdena varierar. Det verkar också som att respondenterna har stött på problem med AI-verktygen tidigare och förväntar sig inte fel fria resultat.

4.2.2 Förväntad ansträngning

Över lag varierar ansträngningsgraden något mellan respondenterna när det kom till att lära sig hantera verktygen. Dock så tycker samtliga respondenter att det i någon grad har varit lätt att lära sig och ansträngningen har varit relativt låg. R1 berättar att tydliga prompts är viktigt när hen tillfrågades om ansträngningar med användandet av AI-verktyg. Detta är ett tema som återkommer och tas upp av R1, R2, R4 och R5.

R2 menar att ansträngningen för att lära sig programmera med AI är relativt lätt. Dock så säger hen att det som är viktigt vid användning av AI-verktyg är att förstå sig på programmering som ämne. Det vill säga ansträngningen för att använda AI verktyg är låg, dock behöver man fortfarande kunskap inom programmering för att det ska fungera väl. Vidare påstår R2 när hen säger att prompta ChatGPT är lätt, dock att det kan vara tidskrävande att skapa bra promptar.

” [...] Jag vet ju bara, om man ger mer information i prompten så kommer det att ge bättre output. Det beror på om man är sugen på att lägga ner tid och lägga på en bra prompt eller inte. ” (Appendix B, #72)

R3 påstår också att det inte är särskilt ansträngande, hen påstår även att man kan komma i gång väldigt snabbt utan någon egentlig förståelse för vad generativ-AI i praktiken är för något. Men detta kommer på gott och ont, eftersom man för första gången kan producera saker i större skala utan att egentligen förstå vad det är, vare sig det är akademiska uppsatser eller kod.

R4 påstår att det svåra för den personen har varit att lära sig prompta ordentligt, men att hen eventuellt undermedvetet blivit bättre på det med tiden, utefter att hen har använt verktyget. Hen påstår även att nu ett år efter att hen började använda AI-verktyg så får man rätt svar mycket snabbare än tidigare.

R4 berättar att hen tyckt det varit ansträngande att lära sig prompta på rätt sätt, R1 menar också att få rätt prompt är den svåra delen med att använda sig av ChatGPT, men antyder

också att det är ganska låg inlärningskurva för att lära sig att prompta. Dock att det är viktigt att förstå outputen så att man vet vad koden gör och hur man ska använda den.

” [...] Det blir mindre ansträngande med AI-verktyg just för informationssökningen och tids-effektiviteten (Appendix A, #28)”

Dock säger R1 också att tiden som respondenten lagt ner på att lära sig Copilot har definitivt varit värt det. Däremot att ChatGPT kan hen ha lagt mycket tid på när det är något som respondenten inte förstår och då kan det ha varit rimligare att lära sig det utan AI-assistansen. Men över lag säger respondenten att inlärningen överlag har varit rimlig kopplat till tiden som lagts ner på att lära sig hantera AI-verktygen.

Även R5 anser att användandet av AI-verktyg är ansträngande ibland, Hen berättar att AI:n kan överkomplicera och ge ett överflödigt svar när hen vill ha en enkelt och snabb lösning. Även om R5 anger någon orsak till detta så berättar hen att det beror på vilken prompt man ger AI: n.

” [...] Man har ju haft ibland några dispyter med den, så att man känner att den inte riktigt fattar vad man vill komma åt, eller att den bara gör det lite för komplicerat än vad det behöver vara (Appendix E, #36)”

Ett annat genomgående tema som alla respondenter vittnar om är AI-hallucinationer eller situationer där AI:n ger felaktiga svar. R1 berättar att hen upplevt AI-hallucinationer som genererat fel kod, däremot nämner R1 att hallucinationen grundar sig i en feltolkning av frågan och kan kopplas till prompting.

” [...] tolka min fråga helt fel och generera fel output (Appendix B #28).

4.2.3 Social påverkan

De flesta respondenterna har i någon grad blivit influerade inom ramen för social påverkan. Framför allt när ChatGPT släpptes fanns det en ”hype” att använda det i de majoriteten av respondenters sociala umgängen. I majoritet av fallen så påverkas respondenterna på så vis att det ses som något positivt att använda. Dock i R4 och R5s fall så fanns det flera inslag av negativitet mot AI användning inom programmering i skolmiljön.

R1 påstår att hen blivit influerad av sociala aspekter på flertalet nivåer, men framför allt av skolan och hur utbildningen är upplagd. Speciellt när det kommer till verktyget Github Copilot men att det från ChatGPT varit mer från vänner och kursare. I början var det också lite av en hype att använda verktyget och R2 påstår vidare att hen blev inspirerad att använda det när hen såg hur snabbt andra kunde lösa vissa uppgifter med verktyget.

R1 exemplifierar känslan när respondenten initialt märkte att folk använde ChatGPT

” [...] Man såg att de kunde göra många grejer med det och så ville man pröva det själv (Appendix A, #44)

R2 förklarar också att hen har blivit socialt påverkad att börja använda ChatGPT och att hen fick det introducerat för sig av en i ett socialt sammanhang och att hen började använda det direkt. Vidare har R2 i enlighet med R1 också börjat använda Github Copilot till följd av att det erbjuds av institutionen. Hen beskriver den allmänna uppfattningen inom hens sociala krets:

” [...] Att den är ett väldigt kraftfullt verktyg att använda. Att man ska använda det och att det är någonting som man måste lära sig om man ska koda med (Appendix B, #102)

R3 påstår att hen började använda sig av ChatGPT direkt när det kom och innan det blev mainstream. Respondenten lärde sig initialt om ChatGPT genom forum på internet där andra teknikintresserade individer befinner sig och där lärde hen sig om lanseringen och började använda sig av det ganska omgående. Dock säger hen att det kan vara tvärtom, eftersom respondenten var tidigt ute så influerade hen snarare andra till att använda det.

R4 säger att hen har blivit influerad av sociala aspekten också, dock är respondenten den enda som blivit influerad mot att AI-assisterad programmering skulle varit negativt. Inledningsvis så blev respondenten influerad av vänner för att börja använda sig av ChatGPT och alla började använda det samtidigt. Dock så har det funnits inslag av personer som tycker att det är fusk i utläringen.

” [...] För att man kan tycka att det är fusk, av princip. Och då blir det lite så att han eller hon vill inte använda det (Appendix D, #59)

Där är det lite mer splittrat i den sociala cirkeln att somliga individer vill använda det och andra vill inte det. R4 säger att hen ofta har den fusk känslan i bakhuvudet när hen gör laborationer och att det känns som att labbarna inte tagit höjd för eventuell användning av AI-verktyg. R5 har också varit med om liknande debatter som varit av en negativ syn på AI, när hen läste institutionen för media och kommunikation hade de lärare som hade negativ inställning mot AI-verktyg i skolan. Respondenten sa dock att detta inte påverkade och att hens inställning till generativ-AI var fortsatt god. Annars har respondenten i likhet med de andra påverkats positivt mot AI-verktyg av både institutionen för informatik och av kurskamrater.

R5 förklarar hur både respondenten och hur kurskamrater uppfattar AI inom programmering

” [...] Jag tror att många ser det som en, som jag också ser, som ett typ av hjälpmedel och ett verktyg för att kunna prestera bättre själv och effektivisera mycket (Appendix E, #44)

4.2.4 Underlättande förhållande

Respondenterna har varierande erfarenheter och åsikter gällande underlättande medel för AI-verktygen ChatGPT och Github Copilot, vilket reflekterar en blandning av stöd och utmaningar från respektive institutioner och individuella preferenser.

Samtliga respondenter medger att det använder ChatGPT-3 i programmeringssyfte, som är gratis att använda. Medan alla har erfarenhet av gratis versionen så anger R1, R2, R3 & R5 att det provat betalversionen GPT-4 för programmering, R2 nämner däremot att hen inte ansåg att betalversionens fördelar var värt pengarna kontra de fördelar som man får:

" [...] Ja, jag ser inte så stor skillnad på 3,5 och 4 till exempel (Appendix B, #130)

R4 som också använder ChatGPT i programmeringssyfte har inte betalat för GPT-4 men anser att betalmodellens användbarhet kan vara värt det:

" [...] Jag har inte betalat för det. Men jag har insett att det kanske kan vara värde. För det verkar vara väldigt användbart (Appendix D, #13)

Github Copilot, som inte har någon gratisversion, uppges av R1, R2 och R3 att erbjuda en studentlicens som R1, R2 & R5 uppges att det tar del av. Både R1 & R2 uppges även att det är just detta AI-verktyg som det använder mest för programmering på grund av att AI-verktyget är integrerat i deras utvecklingsmiljö. Trots att R1 & R2 båda medger att det är verktyget som det använder mest så råder det en tvekan hos båda ifall man hade införskaffat en licens ifall inte institutionen uppmanade och belyste Microsofts studentlicens.

R1 exemplifierar tveksamheten till användandet av betalversioner:

" [...] Jag tror att det hade blivit en barriär där för att det hade tillkommit en månadskostnad för mig för att använda Copilot. Jag kanske hade börjat prenumerera på det någon gång men att det hade varit en mera. Man måste tänka på vad det är värt med pengarna nu när man är student. Så jag tror att jag hade använt det mindre om jag inte hade fått det gratis. Även materialet och videos bidrar också till att vilja använda det." (Appendix A, #62)

R4 använder som enbart gratis versionen av ChatGPT, är medveten att man får använda Copilot via studentlicens men att det inte uppmanas av institutionen och att hen får hör att de inte vill att man ska använda det. Fortsättningsvis belyser hen att studenterna inte får några hjälpmedel i form av dokumentation och instruktionsvideos när det kommer till AI-assisterad programmering alls.

Något som R1 antyder är en av anledningarna till att hen använder det:

" [...] Även materialet och videos bidrar också till att vilja använda det." (Appendix A, #62)

R3 förklarar att hens institution egentligen inte bidrar med licenser till Github Copilot utan att studenter får Copilot via Microsoft, det som institutionen har bidragit med är videor för hur

man ansöker om licens. Vidare för ChatGPT finns det även exempelprompter för ChatGPT berättar R3.

R5 berättar också om att hen har upplevt bidragande faktorer från institutionen med exempelvis promptar inom programmeringskurser som är bra att använda för att lära sig. Under tiden som respondenten studerade programmeringskurserna så fick de inte veta hur Copilot licensen fungerade och respondenten kunde därav inte säga om det finns stöd på institutionen eller inte. Dock torde det vara viktigt med Licens stöd för om institutionen hade bidragit med GPT-4 sa respondenten att hen helt klart hade använt det, trots att respondenten i nuläget tänkte säga upp sin prenumeration på GPT-4.

4.6 Framtiden för Generativ-AI & AI förstärkning

Frågan om framtiden för generativa AI-verktyg såsom ChatGPT och Github Copilot och vilken roll de kommer till att spela i framtidens utveckling var något som alla respondenter hade tankar kring. Alla respondenter var eniga att AI-verktyg kommer fortsätta utvecklas och spela en stor roll i framtidens mjukvaruutvecklare. R1 beskriver AI-verktygen roll i framtidens utveckling som en sorts hjälpreda som tar hand om grundläggande, repetitiva delarna och låta de komplexa delarna skötas av människor. Detta stödjer R2 som menar på att införandet av AI-verktyg kan leda till att färre juniora utvecklare anställs, med ett antal seniora utvecklare som är mer effektiva med hjälp utav AI-verktyg.

R1 och R3 såg potentialen i AI-verktyg för att effektivisera och förbättra kvaliteten på kodning genom att automatisera repetitiva uppgifter och snabba upp konceptförståelsen. R1 ansåg att:

" [...] AI-assisterade verktyg ökar både effektivitet och kodkvalitet genom att automatisera repetitiva uppgifter och snabba upp förståelsen av koncept" (Appendix A, #14).

Dock nämner R1 också att fallgroparna för framtiden kan bli att programmerings studenter inte får den grundläggande förståelsen som är viktig för att kunna använda AI-verktygen på ett bra sätt och att industrin kan bli för beroende av det. R2 reflekterade över att framtida AI-verktyg skulle fokusera på att förbättra förståelsen för kodarkitektur snarare än faktisk kodskrivning, och uttryckte:

" [...] Medan skrivandet av kod kanske inte blir lika viktigt, kommer kunskap om hur saker kopplas samman att vara avgörande" (Appendix B, #36).

R3 betonade att AI-verktyg som Copilot kan göra programmerare mer produktiva på kortare tid, men ersätter inte den mänskliga faktorn helt:

" [...] Även om AI-verktyg som Copilot kan göra programmerare mer produktiva på kortare tid, ersätter de inte den mänskliga faktorn helt" (Appendix C, #52).

Respondenten diskuterade även OpenAI:s vision som är att ersätta allt mänskligt arbete, men att respondenten har svårt att se hur det ska ersätta mänskliga programmerare oavsett hur bra AI blir. Det kan bli så att människan får mer av en redaktörs roll medan AI genererar en stor mängd kod.

R4 uttryckte osäkerhet om framtida AI-verktyg inom programmering och dess effekt på arbetsstrukturen, men noterade potentialen för att förändra balansen mellan juniora och seniora utvecklare:

" [...] AI-verktyg kommer att påverka arbetsstrukturen inom programmering genom att eventuellt minska antalet juniora utvecklare och öka beroendet av seniora utvecklare och AI-baserade verktyg" (Appendix D, #78).

R5 berättade att hen tror och hoppas på att generativ-AI kommer att bli mer väletablerade på företag, både för kvalitetssäkring och för att öka effektivitet. I nuläget ser respondenten positivt på framtiden för generativ-AI dock säger respondenten också i ett citat:

" [...] För ibland märker man ju att shit här är så bra. Och jag kommer aldrig själv kunna komma upp på den nivån. Så det är ju lite skräckblandad och förtjusning med hela AI-verket. Men än så länge ser jag mest positivt på det " (Appendix E, #68).

Samtliga respondenter erkände potentialen i AI-verktyg för att förbättra programmeringsprocessen, även om de uttryckte olika grader av oro för jobbsäkerhet och den mänskliga rollens förändring.

5 Diskussion

Nedan kapitel analyserar empirin som samlats in av respondenterna och för en diskussion kring den. Diskussionen är uppdelat i teman utifrån det UTAUT ramverket likt empirin. Utöver ramverket inkluderas även moderatorn Erfarenhet och Framtid som teman. Först presenteras Erfarenhet, följt av Förväntad prestation, Förväntad ansträngning, Social påverkan, Underlättande medel och avslutningsvis Framtid.

5.1 Erfarenhet

Erfarenheten spelar en avgörande roll som moderator inom ramverket UTAUT (Venkatesh et al. 2003). Huvudsakligen på grund av att moderatoren påverkar tre utav modellens fyra konstrukt, förväntad ansträngning, social påverkan och underlättande medel. Studien syftade därför till att utforska respondenternas erfarenhet av användandet av AI-verktyg i programmeringssyfte.

Erfarenhetens roll i att öka förtroendet med teknologi är central i UTAUT-modellen, där regelbunden användning av teknologi leder till minskad upplevd komplexitet och ökad användaracceptans (Venkatesh et al. 2003). Studiens empiri visar på att respondenterna R1 och R2 och R3 som har använt AI-verktyg som GitHub Copilot och ChatGPT under en längre tid, utvecklat en djupare förståelse för dessa verktygs kapaciteter och begränsningar. Denna ökade förtroende har gjort det möjligt för dem att effektivare navigera och utnyttja verktygens potential i sina programmeringsprojekt.

Denna förtroende speglas även i deras förmåga att anpassa sig till verktygens specifika output och att kritiskt bedöma och välja vilka förslag som ska implementeras i deras kod. Genom deras erfarenhet har de blivit mer bekväma med att använda dessa AI-verktyg, vilket minskar den osäkerheten som ofta associeras med nya teknologier. Deras erfarenhet underlättar en snabbare och mer effektiv integration av tekniken i vardagliga programmeringsprocesser, vilket stärker teknikens acceptans.

Respondenternas förmåga att inte bara anpassa sig till utan även att innovera med AI-verktygen framhäver ytterligare aspekten av erfarenhet. Enligt Brynjolfsson och McAfee (2017) möjliggör erfarenhet att användare inte bara uppfattar teknologi som ett hjälpmedel för grundläggande uppgifter utan också som ett verktyg för att utforska nya, innovativa användningsområden. Denna aspekt av erfarenhet kan direkt kopplas till hur R1, R2 och R3 använder AI-verktygen för att utforska och implementera nya lösningar som förbättrar deras kodningsprocesser.

Genom respondenternas fortsatta användning av AI-verktyg har de kunnat identifiera unika tillämpningar som går utöver det vanliga tillämpningsområden inom ramen för programmering. Dessa kan inkludera komplexa problemupplösningar och automatisering av uppgifter

som tidigare krävde omfattande manuellt arbete. Detta inte bara ökar produktiviteten utan även bidrar till en kontinuerlig förbättring av kodkvaliteten och utvecklingsprocesserna inom deras organisationer.

Det påpekas också av Dakhel et al. (2023) att erfarenhet är avgörande för hur väl utvecklare kan utnyttja AI-verktyg och hantera eventuella felaktiga eller suboptimala kodförslag. Samtliga respondenter började använda AI-verktygen, särskilt ChatGPT och GitHub Copilot, strax efter deras lansering. Detta kan indikera att utvecklare med tidigare erfarenhet och en uppdaterad förståelse för den senaste teknologin är mer benägna att snabbt anamma och integrera dessa verktyg i sin arbetsprocess, vilket stöder Dakhels poäng om att erfarenhet kan vara avgörande för att omfamna nya AI-verktyg. Liksom Dakhel et al. (2023) påpekar respondenterna att de har erfarenhet av felaktiga eller vilseledande kodförslag från AI-verktygen. Detta understryker behovet av erfarenhet för att kunna kritiskt granska och korrigera sådana fel för att undvika att verktygen leder användaren åt fel håll.

5.2 Acceptans

5.2.1 Förväntad prestanda

Förväntad prestanda är nästa konstrukt i UTAUT som relaterar till den grad till vilken en användare tror att användningen av tekniken kommer att hjälpa dem att förbättra sin prestation i arbetet (Venkatesh et al. 2003). Respondenterna i empiristudien uttryckte en stark uppfattning om att användningen av AI-verktygen Github Copilot och ChatGPT ökade deras effektivitet och produktivitet inom programmering. Denna uppfattning speglar litteraturens påståenden, som vi ser i Zhang et al. (2023) och Kuhail et al. (2024), där AI-verktyg beskrivs som effektiva i att automatisera och optimera programmeringsprocessen.

Den generella uppfattningen var att AI-verktyg var bra inom specifika områden i mjukvaruutvecklingsprocessen, exempelvis för att felsöka eller generera specifik kod i syfte att underlätta för respondenterna att jobba vidare. Detta stämmer väl överens med vad Hassani et al. (2020) säger och antyder att AI är till för att förstärka det mänskliga arbetet och Brynjolfsson & McAfee (2017) som menar på att AI är som mest effektiv när den automatiserar specifika steg i en process.

Respondent R1 & R3 rapporterade specifikt hur AI-verktygen hjälpte dem att snabbt generera kod och förstå nya programmeringskoncept, vilket direkt ökade deras prestanda genom att minska den tid de annars hade spenderat på dessa uppgifter. Detta stöds av Zhang et al. (2023) som påpekar att AI-verktyg kan minska tidsåtgången för kodning avsevärt genom förslag och autokomplettering, vilket inte bara snabbar upp processen utan också minskar felmarginalen.

Respondent R2 nämnde även hur dessa verktyg hade positiv påverkan på deras förmåga att felsöka och lära sig nya tekniska färdigheter. Denna upplevelse av förbättrad problemlösningsförmåga med AI-verktygens hjälp är i linje med Kuhail et al. (2024), som framhäver att

AI-integrationen i utvecklingsprocesser kan leda till djupare förståelse och innovativ tillämpning av kod. Detta var även generellt applicerbart för respondenterna som använde AI-verktyg i olika utsträckning för att lära sig om programmeringsspråk.

Respondenterna har dock stött på svårigheter när de använder sig av AI-verktyg i form av att de gått runt i cirklar eller att de har upplevt AI-hallucinationer. Enligt Naveed et al. (2023) och Shen et al (2023) är AI hallucinationer en farhåga som tillkommer när generativ-AI genererar trovärdiga svar som i verkligheten är felaktiga. Detta kan kopplas till empirin eftersom majoriteten av respondenterna har upplevt det och i någon utsträckning måste granska sina svar ut efter risken för AI-hallucinationer.

Samtidigt uttryckte respondenterna en känsla av att AI-verktygens förmåga att underlätta inlärning och förståelse ökade deras övergripande kompetens inom programmering, vilket pekar på en signifikant förbättring i deras arbetseffektivitet. Detta stöder idén att förväntad prestanda inte enbart handlar om den omedelbara effektiviteten, utan också om hur tekniken bidrar till användarens långsiktiga utveckling.

5.2.2 Förväntad ansträngning

Inom UTAUT är förväntad ansträngning den ansträngning som individerna förväntar vid användandet av teknologin. Utifrån empirin är det tydligt att samtliga respondenter tycker att det är lätt att använda AI-verktyg. Kuhail et al. (2024) betonar hur AI-verktyg för assisterad programmering har revolutionerat mjukvaruutvecklingsprocessen genom att automatisera och optimera olika aspekter av kodskrivning. Deras arbete ger en inblick i hur dessa verktyg har minskat den manuella ansträngningen genom att automatisera monotona aktiviteter. Detta resonemang stöds av empirin, där respondenterna tydligt framhäver fördelarna med AI-verktyg för att minska ansträngningen vid kodskrivning. Till exempel, respondent R1 delar erfarenheten av att AI-verktygen har effektiviserat processen och minskat den manuella arbetsbördan genom att jämföra det med att skriva på papper jämfört med att skriva i Word. Liknande observationer görs av respondent R4, som noterar att de får svar mycket snabbare nu än tidigare genom att använda AI-verktyg. Dessa erfarenheter bekräftar att AI-verktygen verkligen minskar den manuella ansträngningen och effektiviserar kodskrivningsprocessen.

Det svåra verkar vara att lära sig prompta så att man får en bra output från AI verktygen. Men i stort påstår respondenterna att man kan sätta i gång utan att ha några förkunskaper om generativ-AI. Kuhail et al. (2024) nämner i sin forskning att AI-verktyg för assisterad programmering kan kräva en förståelse för hur man skapar lämpliga prompts för att få önskade resultat. Detta resonemang stödjer upplevelserna i empirin där respondenterna uttrycker svårigheter med att få korrekta resultat från AI-verktygen om de inte formulerar sina frågor eller prompts på rätt sätt. Detta indikerar att felaktiga eller inadekvata prompts kan resultera i felaktiga svar från AI-verktygen, vilket i sin tur skapar en extra ansträngning för användarna att korrigera och anpassa sin input för att få önskade resultat.

Dock säger R2 att det fortfarande är viktigt att förstå sig på programmering för att AI-verktyg ska vara av någon egentlig nytta. Detta kan kopplas till argumentet av Dakhel et al. (2023) som betonar att AI verktyg är särskilt fördelaktiga när de används av erfarna programmerare.

Som tidigare nämnts så är AI-verktygen som finns på marknaden idag väldigt lätta att sätta sig in i menar samtliga respondenter. Generellt tyckte respondenterna att ansträngning kring både informationssökning och direkt relaterade problem till programmering som felsökning minskade med AI-verktyg. I mångt och mycket säger respondenterna i likhet med R1 att det har varit värt den nerlagda tiden har varit värt det. Venkatesh et al. (2003) antyder att den förväntade ansträngningen som individer lägger ner på ett nytt IS väger in i deras användnings beteende, generativ-AI är så pass lätt att använda att de flesta inte reflekterar mycket över svårigheterna.

Respondenterna i studien betonar också att det finns vissa ansträngande delar, det kan exempelvis vara att det är svårt att få en korrekt output eller att koden blir för komplex för syftet. Dessa utmaningar är något som Dakhel et al. (2023) lyfter när det nämns att AI kan generera suboptimal kod, vilket kan vara problematiskt för nybörjare. Detta kan också vara en av anledningarna till att respondenterna använder AI-verktyg varsamt, kopplat till kodgenerering, när det handlar om att lära sig nya programmeringsspråk. Detta stämmer även överens med Hamza et al. (2023) som betonar vikten av att AI-verktyg ger förslag medan en utvecklare fortfarande måste kontrollera outputen.

Forskningen av Naveed et al. (2023) och Shen et al. (2023) belyser farhågor kring AI-hallucinationer, där språkmodeller kan generera svar som verkar trovärdiga men är felaktiga. Denna aspekt kan kopplas till empirin genom att identifiera situationer där respondenterna upplever att AI-verktygen ger felaktiga eller missvisande kodförslag, vilket i sin tur skapar en extra ansträngning för dem att korrigera eller förstå den genererade koden.

I empirin nämner respondenterna, särskilt R1, R2, R4 och R5, att de har erfarenhet av AI-hallucinationer eller situationer där AI-verktygen genererar felaktig kod. Till exempel, R1 berättar om att hen har upplevt AI-hallucinationer där kod genererats felaktigt på grund av en felaktig tolkning av frågan. Detta indikerar att farhågor om AI-hallucinationer kan återspeglas i respondenternas upplevelser vid användning av AI-verktyg i programmering (Naveed et al. 2023).

5.2.3 Socialt inflytande

Respondenternas erfarenheter av att bli influerade av sociala nätverk och grupper i användningen av AI-verktyg för programmering kan analyseras genom ramverkets tredje kärnkonstrukten, socialt inflytande, inom UTAUT (Venkatesh et al. 2003). Enligt UTAUT påverkas individers antagande och användning av ny teknik av sociala faktorer, inklusive påtryckningar från vänner, kollegor och sociala nätverk. Detta resoneras väl med respondenternas berättelser om att deras introduktion till AI-verktyg ofta kom genom sociala interaktioner, såsom påverkan från skolan, vänner och andra i deras sociala kretsar. Dessutom betonar Wang et al.

(2020) betydelsen av att analysera samarbetet mellan människor och AI med hänsyn till sociala normer och förväntningar för att förstå hur dessa faktorer formar vår syn på teknologin. Exempelvis beskriver respondent R1 hur hen inspirerades att använda AI-verktyg, såsom ChatGPT och Github Copilot, av skolan och vänner. Liknande berättelser finns hos R2 och R3, där introduktionen till AI-verktyg kom genom sociala sammanhang och där hype och positiv förstärkning spelade en central roll. Å andra sidan finns det också exempel på negativa attityder och debatter kring användningen av AI-verktyg, särskilt hos respondenterna R4 och R5. Deras erfarenheter av att ifrågasätta användningen av AI-verktyg, även inom en miljö där det är vanligt förekommande, stödjer Wang et al. (2020) och Hamza et al. (2023) argument om vikten av att förstå både potentialen och begränsningarna hos dessa verktyg.

Vidare kan respondenternas upplevelser av social påverkan kopplas till litteraturen som presenteras i kapitlet om AI-assisterad mjukvaruutveckling av Kuhail et al. (2024). Kuhail et al. (2024) belyser hur social interaktion och påverkan spelar en betydande roll i antagandet och användningen av AI-verktyg inom mjukvaruutveckling. Dessa verktyg introducerades och användes av respondenterna på grund av påverkan från deras sociala miljö, vilket indikerar en djupgående samverkan mellan teknologi och sociala strukturer.

Fortsättningsvis så visar respondenternas svar hur deras användning av AI-verktyg för programmering påverkades av sociala influenser från olika delar av deras liv. På arbetsplatsen, i skolan och inom deras sociala kretsar blev de introducerade till och influerade att använda dessa verktyg av kollegor, lärare och vänner. Detta stöder Venkateshs idé om att socialt inflytande spelar en avgörande roll i teknologianslutning och användning (Venkatesh et al 2003) På liknande sätt utforskar Hamza et al. (2023) hur AI-verktyg har utvecklats till att bli mer än bara assistenter, men faktiska samarbetspartners, vilket speglar sig i respondenternas berättelser om att förlita sig på AI för mer än bara rutinmässiga uppgifter.

5.2.4 Underlättande förhållande

Underlättande förhållanden inom UTAUT-modellen handlar om den grad till vilken en individ tror att en organisation eller infrastruktur stödjer användningen av ett system. Enligt Venkatesh et al. (2003), påverkar dessa förhållanden direkt användningens frekvens och intensitet av en teknologi. I resultatet av studien framgår det att verktyg som ChatGPT, inte bara används för programmeringsuppgifter utan också som pedagogiska verktyg som faciliteter inlärning och förståelse av nya koncept inom programmering, vilket stötts av Nhavkar & Goel (2023) som betonar generativ-AI:s förmåga att underlätta inlärning och felsökning inom programmering, vilket bidrar till ökad effektivitet och stöd i utbildningskontexter.

Utifrån empirin tycks respondenterna positivt inställda till användningen av AI-verktyg, inte bara som en del av deras programmeringsarbete utan även som en integrerad del av deras inlärningsprocess. Respondenterna rapporterar att användningen av sådana verktyg minskar den tid och ansträngning som krävs för att lösa programmeringsproblem och förstå nya programmeringsramverk. Denna upplevelse av AI-verktyg som en pedagogisk resurs matchar väl med argumenten som Nhavkar & Goel (2023) presenterar.

Institutionellt stöd och utbildning är avgörande för att effektivt integrera AI-verktyg som ChatGPT och Github Copilot i programmeringsutbildning och arbetsmiljöer. Respondenterna i empirin betonar vikten av att institutionen tillhandahåller adekvat stöd och utbildning för att maximera fördelarna med dessa verktyg. Till exempel påpekar R3 att institutionen erbjuder tillgång till Github Copilot via Microsoft, och att de även tillhandahåller videor för hur man ansöker om licenser. Detta visar på institutionens roll i att underlätta tillgången till AI-verktyg för studenter och att främja deras effektiva användning. Likaså belyser R1 vikten av institutionellt stöd genom att reflektera över huruvida hen skulle ha använt betalversionen av ChatGPT om den inte tillhandahölls gratis av institutionen. Detta antyder att institutionell support kan påverka användarnas beslut att använda och dra nytta av AI-verktyg.

För att ytterligare diskutera underlättande förhållanden, beskriver R1 hur integreringen av ChatGPT i undervisningen har möjliggjort snabbare förståelse av komplexa koncept. Detta illustrerar direkt hur AI-verktyg kan fungera som en del av den infrastruktur som stödjer effektiv användning av teknologi i utbildningsmiljöer. R1 påpekar att med hjälp av ChatGPT kan hen snabbt få förklaringar till komplicerade problem, vilket inte bara sparar tid utan också förbättrar kvaliteten på koden, anger R1.

R3 delar liknande åsikter och betonar hur användningen av Github Copilot har förbättrat produktiviteten genom att automatisera generering av boilerplate-kod, vilket gör det möjligt för dem att fokusera på mer komplexa aspekter av programmeringsprojekt. R3 uttrycker att Github Copilot har hjälpt hen att effektivisera utvecklingsprocessen genom att ta hand om repetitiva uppgifter, vilket gör att respondenten kan använda tid till att tackla svårare utmaningar. Detta går i enlighet med vad Brynjolfsson & McAfee (2017) menar på när de säger att AI är som mest effektiv när den används för att automatisera specifika steg i en process. Detta möjliggör för människor att fokusera på mer komplexa och kreativa aspekter av uppgifter.

Dessa observationer visar på vikten av att institutioner och utbildningsanordnare erkänner och stödjer användningen av dessa verktyg. Det krävs en infrastruktur som inte bara tillhandahåller tillgång till dessa verktyg utan också utbildar användarna i hur de ska användas effektivt och ansvarsfullt.

5.3 Framtid

Respondenterna diskuterade samtliga framtidsutsikter, vissa tog i enlighet med Hassani et al. (2020) ståndpunkten att det kommer bli mer AI-förstärkning i framtiden och utvecklare kommer att bli förstärkta av AI snarare än utbytta. Dock fanns det orors moment som kan kopplas mer till det andra AI lägret som beskrivs i Hassani et al. (2020) där AI påstås kunna ersätta vissa processer helt. Men som forskningen av Brynjolfsson och McAfee (2017) visar så ersätter sällan AI hela jobb utan fungerar bäst som komplement utan kompletterar snarare processer. Många av respondenterna beskriver AI som exempelvis R1 som en hjälpredda eller något som kommer användas för att utföra repetitiva uppgifter inom programmering. Detta stämmer

bra i enlighet med Brynjolfsson och McAfee (2017) som säger att AI är som mest effektiv när den används för specifika steg i en process.

Från empirin tyder som sagt mycket på att respondenterna tror ganska starkt på AI-förstärkelse i framtiden men en viss rädsla för att AI kommer att automatisera stora delar av mjukvaruutvecklingen visade sig också. Bland annat R5 som säger att användningen av AI kan komma med skräckblandad förtjusning kopplat till hur bra exempelvis ChatGPT är på att generera kod. Detta är något som delas mellan många av respondenterna och R4 säger att det i framtiden kanske kommer behövas mindre juniora utvecklare och ett företag i stället blir beroende av seniora utvecklare med god kunskap av AI-verktyg. Detta kan dock bero på det som Mitchell (2021) talar om kring AI:s optimism, just nu är vi inne i en period av stor optimism kring AI och speciellt kring generativ-AI. Men som Mitchell (2021) beskrev så följer dessa cyklar ofta av så kallad AI vinter. Dock är det inte heller helt orimligt som R4 säger, AI har som tagits upp av Hassani et al. (2020) och Brynjolfsson och McAfee (2017) och beskrivits som ett verktyg som kan effektivisera processer. Om de seniora utvecklarna blir extremt effektiva med hjälp av AI verktyg kan det bli mindre utvecklare som behövs i framtiden. Vidare menar även Van Slyke et al. (2023) på att AI-verktyg gör så att ribban höjs för framtida utvecklare och man måste bli mer tekniskt kunnig som utvecklare, vilket stämmer överens med framtiden som många av respondenterna förväntade sig.

Det är viktigt att också beakta det perspektiv som respondenterna delade om både positiva och negativa aspekter av AI-assisterad programmering. Även om respondenterna uttryckte en viss oro för automatisering av vissa arbetsuppgifter med hjälp av AI, visade de också en positiv syn på AI:s potential att förbättra kodningsprocessen och öka effektiviteten.

Sammanfattningsvis pekar respondenternas erfarenheter och uppfattningar, tillsammans med forskningen av Hassani et al. (2020), Brynjolfsson och McAfee (2017), samt Özpolat et al. (2023), på en komplex framtid för AI-assisterad programmering, där användningen av AI-verktyg förväntas öka samtidigt som det finns oro för dess potentiella inverkan på arbetsstrukturen och jobbsäkerheten för vissa roller inom mjukvaruutveckling. Det är viktigt att fortsätta utforska och övervaka denna utveckling för att förstå dess långsiktiga konsekvenser. I linje med Mitchell (2021) kan det också vara klokt att beakta historiska mönster av teknologiska cykler, där perioder av stor optimism kring AI ofta följs av en så kallad "AI-vinter". Denna insikt påminner oss om att det är viktigt att vara realistisk om AI:s utveckling och att vara beredd på eventuella utmaningar och svängningar i teknologins framtid.

6 Slutsats

Genom att undersöka ”*Vad är utvecklarens erfarenhet och acceptans av AI-verktyg för mjukvaruutvecklingsprocesser inom ramen för programmeringskurser*” har denna studie haft som mål att bidra till en fördjupad förståelse för användningen av teknologi inom denna kontext.

Det författarna har funnit i empirin och diskussionen är att det finns en bred acceptans av användandet av AI-verktyg, såsom ChatGPT och Github Copilot bland utvecklarna. AI-verktyg ses som ett kraftfullt hjälpmedel som underlättar programmeringen för utvecklarna i ramen för programmeringskurser. Resultaten tyder på att erfarenhet spelar en avgörande roll för utvecklarnas förtrogenhet och acceptans av AI-verktyg. Deras erfarenheter visar hur kontinuerlig användning av teknologi leder till ökad förtrogenhet och förmåga att effektivt utnyttja verktygens potential. Dessutom understryker studien betydelsen av förväntad prestanda, där användarna upplever att AI-verktyg ökar deras produktivitet och effektivitet inom programmering. Vad gäller den förväntade ansträngningen av användandet framgår det i studien att användarna ser en viss ansträngning i form av prompting och AI-hallucinationer som producerar felaktiga och missledande svar. Respondenterna i studien antyder dock att AI-verktygen fortfarande ger en ökad produktivitet och effektivitet, vilket indikerar att de ser en rimlig förväntad ansträngning jämfört med den förväntade prestandan för att använda dessa AI-verktyg.

Socialt inflytande har också visat sig vara en betydande faktor i användningen av AI-verktyg för mjukvaruutveckling. Respondenternas svar visar hur deras introduktion och användning av dessa verktyg påverkades av sociala interaktioner och påverkan från olika delar av deras livsmiljö. Sammanfattningsvis ser författarna att det övervägande positiva sociala inflytandet har bidragit till en god inställning till AI-verktygen där utvecklarna är benägna att själva använda det.

Likaså har underlättande förhållanden, såsom tillgång till adekvat infrastruktur och stöd från organisationer och utbildningsanordnare, visat sig vara avgörande för att främja användningen av AI-verktyg inom utbildningsmiljöer. Respondenternas erfarenheter visar att institutionellt stöd och utbildning är avgörande för att effektivt integrera AI-verktyg i programmeringsutbildning. Tillgång till dessa verktyg via studentlicenser och stödresurser underlättar användarnas förmåga att dra nytta av verktygens potential.

Vidare diskuteras framtidsutsikter för AI-assisterad programmering, där både optimism och oro för teknologins utveckling framkommer. Även om respondenterna tror på AI:s potential att förbättra kodningsprocessen genom att automatisera och öka effektiviteten, finns det också en viss oro och medvetenhet för dess potentiella inverkan på arbetsstrukturen inom mjukvaruutveckling.

Genom att integrera teoretiska begrepp från UTAUT-modellen och tidigare forskning har denna studie bidragit till en fördjupad förståelse för användningen av AI-verktyg inom mjukvaruutveckling och öppnat upp för vidare forskning och diskussion kring detta ämne.

6.1 Kritisk inblick

6.1.1 Studiens skala & urval av respondenter

Denna studie är ganska smal till skalan och utförd på studenter som för visso programmerar mycket men inte lika mycket som programmerare inom arbetslivet gör. Detta leder till att uppfattningen och acceptansen kan variera mycket om studien i stället hade utförts på yrkesverksamma. Resultatet utifrån det perspektivet hade kunnat vara mer givande eftersom yrkesverksamma har mer erfarenhet av programmering.

6.1.2 Författarnas Bias

Författarna i denna studie är medvetna om vikten att identifiera och hantera eventuella bias som kan påverka tolkningen av den empiriska datan och analysen. Författarna har en positiv syn på generativ-AI och dess tillämpning inom mjukvaruutveckling. Detta innebär att författarnas egna åsikter kan ha påverkat både genomförande av intervjuer och tolkning av datan.

För att minimera risken för bias har författarna strävat efter att vara medvetna om sina egna förutfattade meningar och inte låta dem speglas i intervjuer eller i diskussionen. Under samtliga intervjuer har båda författarna deltagit för att bidra till en ömsesidig reflektion och minska risken för en ensidig tolkning.

Trots författarnas medvetna ansträngning kan bias inte uteslutas helt och det är därför viktigt för läsaren att var medveten om författarnas ståndpunkt och att resultatet bör tolkas med detta i åtanke.

6.1.3 Empiriskt material

Avslutningsvis skulle det ha varit fördelaktigt att utföra mer djupgående intervjuer med mer teoretiskt kopplade och nyanserade frågor. Genomsnittslängden på våra intervjuer var 28 minuter, vilket kan innebära att vår empiri kan betraktas som något begränsad. Även om vi intervjuade totalt fem respondenter, vilket författarna ansåg tillräckligt i en kvalitativ ansats som vår, skulle det eventuellt ha varit mer givande med mer ingående intervjufrågor.

6.2 Förslag till vidare forskning

AI-assisterad programmering är ett område som utvecklas och förändras väldigt fort, Målet med denna studie har varit att försöka öka förståelsen för mjukvaruutvecklarens erfarenheter och acceptans av AI-verktyg inom mjukvaruutvecklingsprocesser i en universitetsmiljö, med hjälp av ramverket för Unified Theory of Acceptance and Use of Technology (UTAUT). För att fortsätta bidra till forskningsområdet och bredda förståelsen för användningen av AI-verktyg inom mjukvaruutveckling, föreslås det att replikera studien i en professionell arbetsmiljö,

där mjukvaruutvecklare arbetar i branschen, skulle ge ytterligare insikter i användningen av AI-verktyg. Detta skulle möjliggöra för forskare att undersöka hur användningen av AI-verktyg påverkas av den ökade pressen på att vara effektiv och produktiv i en arbetsmiljö där resultatet har direkta konsekvenser för företaget. Genom att studera hur AI-verktygen integreras i professionella arbetsflöden och hur de påverkar utvecklarnas produktivitet kan man få en mer nyanserad förståelse för deras användning och inverkan. En annan aspekt som är värd att utforska är användarnas förtroende och uppfattningar om AI-verktygens tillförlitlighet. Det är av betydelse att undersöka hur användarna upplever AI-verktygens förmåga att producera korrekt kod och hur detta påverkar deras tilltro till verktygen. Genom att förstå användarnas uppfattningar och eventuella utmaningar kan utvecklare arbeta för att förbättra användarupplevelsen och öka förtroendet för AI-verktygen. Detta kan i sin tur leda till ökad acceptans och användning av AI-verktyg inom mjukvaruutvecklingsbranschen.

Appendix A – G removed

Appendix H - AI-bidragsredogörelse

Studien har använt sig av AI-verktygen ChatGPT, Bing Copilot och WhisperAI. WhisperAI har använts som stöd för transkribering av samtliga intervjuer. Samtliga intervjuer har använt WhisperAI lokalt med ”medium” modellen och på svenska för att transkribera MP4A filerna som sedan granskades av författarna. I kapitel 3.3.5 förklaras användningen av WhisperAI vid automatisktranskribering mer ingående. Det andra två AI-verktygen, ChatGPT-4 och Bing Copilot har använts kontinuerligt under studiens gång för att spåna idéer, agera bollplank för potentiella idéer, sammanfattning av artiklar för att få en överblick ifall artikeln är värd att läsa samt till förbättring av struktur och meningsbyggnad i text. Båda AI-verktygen har även använts för kodning av transkriberingarna där AI-verktygen fick analysera de som ansågs viktigt för att författarna sedan kunde undersöka om deras egen analys missat något

7 Referenser

- Agarwal, M. & Saxena, A., (2019). An Overview of Natural Language Processing. *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 7, no. 5, pp. 2810-2813, <https://doi.org/10.22214/ijraset.2019.5462>
- Bandi, A., Adapa, P. V. S. R. & Kuchi, Y. E. V. P. K., (2023). The Power of Generative AI: A Review of Requirements, Models, Input–Output Formats, Evaluation Metrics, and Challenges. *future internet*, vol. 15, no. 8, pp. 260-320, <https://doi.org/10.3390/fi15080260>
- Bell, E., Bryman, A. & Harley, B. (2019). *Business Research Methods*. Oxford: Oxford University Press.
- Blut, M., Chong, A., Tsiga, Z. & Venkatesh, V. (2021). Meta-Analysis of the Unified Theory of Acceptance and Use of Technology (UTAUT): Challenging Its Validity and Charting a Research Agenda in the Red Ocean, Social Science Research Network, <https://doi.org/10.17705/1jais.00719>.
- Sogeti VINT. (2014). The Fourth Industrial Revolution. Things to Tighten the Link Between IT and OT [pdf], <https://www.sogeti.com/globalassets/global/special/sogeti-things3en.pdf>
- Bostrom, N. (2020). *Superintelligens, Vägar, faror, strategier*. Oxford: Oxford University Press, Cop.
- Booch, G. (2018). The History of Software Engineering, *IEEE Software*, vol. 35, no. 5, pp.108–114, <https://doi.org/10.1109/MS.2018.3571234>
- Brynjolfsson, E., Li, D. & Raymond, R. L. (2023). Generative AI at Work, preprint, pp. 1-56, <https://arxiv.org/abs/2304.11771>
- Brynjolfsson, E. & McAfee, A. (2017). The Business Of Artificial Intelligence. *Harvard Business Interview*, no. 7, pp. 3-11, <https://starlab-alliance.com/wp-content/uploads/2017/09/The-Business-of-Artificial-Intelligence.pdf>
- Cambria, E. & White, B. (2014). Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine*, vol. 9, no. 2, pp. 48-57, <https://doi.org/10.1109/MCI.2014.2307227>.
- Clarke, P. & O'Connor, R. V. (2012). The situational factors that affect the software development process: Towards a comprehensive reference framework. *Information and Software Technology*, vol. 54, no. 5, pp. 433-447, <https://doi.org/10.1016/j.infsof.2011.12.003>
- Chui, M., Hazan, E., Roberts, R., Singla, A., Smaje, K., Sukharevsky, A., Yee, L. & Zimmel, R. (2023). Economic Potential of Generative AI | McKinsey, [www.mckinsey.com](https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier#key-insights), Available Online: <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier#key-insights>
- Dwivedi, Y. K., Hughes, L., Ismagilova, E., Aarts, G., Coombs, C., Crick, T., Duan, Y., Dwivedi, R., Edwards, J., Eirug, A., Galanos, V., Ilavarasan, P. V., Janssen, M., Jones, P., Kar, A. K., Kizgin, H., Kronemann, B., Lal, B., Lucini, B. & Medaglia, R. (2021).

- Artificial Intelligence (AI): Multidisciplinary Perspectives on Emerging Challenges, Opportunities, and Agenda for Research, Practice and Policy, *International Journal of Information Management*, vol. 57, <https://doi.org/10.1016/j.ijinfomgt.2019.08.002>.
- Dwivedi, Y. K., Rana, N. P., Jeyaraj, A., Clement, M. & Williams, M. D. (2019). Re-Examining the Unified Theory of Acceptance and Use of Technology (UTAUT): Towards a Revised Theoretical Model, *Information Systems Frontiers*, vol. 21, no. 3, pp.719–734, <https://doi.org/10.1007/s10796-017-9774-y>.
- Github. (2024). Copilot: The world’s most widely adopted AI developer tool, <https://github.com/features> [Accessed 10 April]
- Hamza, M., Siemon, D., Akbar, M. A. & Rahman, T. (2023). Human AI Collaboration in Software Engineering: Lessons Learned from a Hands on Workshop, ACM woodstock conference proceedings, <https://doi.org/10.48550/arXiv.2312.10620> [Accessed 10 March]
- Hassani, H., Silva, E. S., Unger, S., TajMazinani, M. & Feely, S. M. (2020). Artificial Intelligence (AI) or Intelligence Augmentation (IA): What Is the Future?, *AI 2020*, vol. 1, no. 2, pp.143–155, <https://doi.org/10.3390/ai1020008>
- IBM. (2024). What is software development?, <https://www.ibm.com/topics/software-development> [Accessed 31 March]
- Johnson, P. C., Laurell, C., Ots, M. & Sandström, C. (2022). Digital Innovation and the Effects of Artificial Intelligence on Firms’ Research and Development – Automation or Augmentation, Exploration or Exploitation?, *Technological Forecasting and Social Change*, vol. 179, p.121636, <https://doi.org/10.1016/j.techfore.2022.121636>.
- Jordan, M. I. & Mitchell, T. M. (2020). Machine Learning: Trends, Perspectives, and Prospects, *Science*, vol. 349, no. 6245, pp.255–260, <https://www.science.org/doi/10.1126/science.aaa8415>.
- Karolinska Institutet, Universitetsbiblioteket, Grå litteratur, <https://kib.ki.se/soka-vardera/gralitteratur> [Accessed
- Kuhail, A. M., Taher, F. & Barengueres, J. (2024) Integrating Generative AI Into the Design Process: A Case Study on Space Haptic Boots, AHFE, 2024 Proceedings, https://www.researchgate.net/publication/377567786_Integrating_Generative_AI_into_the_Design_Process_A_Case_Study_on_Space_Engineering#full-text
- Mahesh, B. (2020). Machine Learning Algorithms -A Review. *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381-386, <https://doi.org/10.21275/ART20203995>.
- McAfee, A., Rock, D. & Brynjolfsson, E. (2023). How to Capitalize on Generative AI. *Harvard Business Review*, vol. November-December, <https://hbr.org/2023/11/how-to-capitalize-on-generative-ai>
- Mitchell, M. (2021). Why AI Is Harder than We Think, Available Online: <https://arxiv.org/abs/2104.12871>
- Nah, F. F.-H., Zheng, R., Cai, J., Siau, K. & Chen, L. (2023). Generative AI and ChatGPT: Applications, Challenges, and AI-Human Collaboration, *Journal of information technology case and application research*, vol. 25, no. 3, pp.1–28, <https://doi.org/10.1080/15228053.2023.2233814>.

- Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N. & Mian, A. (2023). A Comprehensive Overview of Large Language Models, preprint, <https://arxiv.org/abs/2307.06435>.
- Nhavkar, K. V. & Goel, K. D. S. (2023). Impact of Generative AI on IT Professionals. *International Journal for Research*, vol. 11, no. 7, <https://doi.org/10.22214/ijraset.2023.54515>
- Moradi Dakhel, A., Majdinasab, V., Nikanjam, A., Khomh, F., Desmarais, M. C. & Jiang, Z. M. (2023). GitHub Copilot AI pair programmer: Asset or Liability?. *The Journal of Systems & Software*, vol. 203, <https://doi.org/10.1016/j.jss.2023.111734>
- Oates, B. J., Griffiths, M. & McLean, R., 2022. *Researching Information Systems and Computing*. London: Sage Publications Ltd.
- OECD.ai (2024) OECD AI Principles overview, <https://oecd.ai/en/ai-principles> [Accessed 1 may 2024]
- OpenAI. (2022) Introducing Whisper, <https://openai.com/index/whisper/> [Accessed 5 may 2024]
- Rempel, G. H. & Mellinger, M. (2015). Bibliographic Management Tool Adoption and Use A Qualitative Research Study Using the UTAUT Model. *Reference and User Services Association*, vol. 54, no. 4, <https://doi.org/10.5860/rusq.54n4.43>.
- Reinecker, L. Stray Jørgensen, P. & Hedelund, L. (2014), *Att skriva en bra uppsats*, Stockholm: Liber
- Roumeliotis, K. I. & Tselikas, N. D. (2023). ChatGPT and Open-AI Models: A Preliminary Review. *future internet*. vol. 15, no. 6, <https://doi.org/10.3390/fi15060192>
- Russel, J. S. & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach*, 4th edn. New Jersey: Pearson.
- Salesforce. (2023). Top Generative AI Statistics for 2023, <https://www.salesforce.com/au/news/stories/generative-ai-statistics/>, [Accessed 1 April 2024]
- Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, vol. 2, no. 160, pp. 1-21, <https://doi.org/10.1007/s42979-021-00592-x>
- Shen, Y., Heacock, L., Elias, J., Hentel, K. D., Reig, B., Shih, G. & Moy, L. (2023). ChatGPT and Other Large Language Models Are Double-Edged Swords, *Radiology*, vol. 307, no. 2, <https://doi.org/10.1148/radiol.230163>
- Slyke, C. V., Johnsson, R. D. & Sarabadani, J. (2023). Generative Artificial Intelligence in Information Systems Education: Challenges, Consequences, and Responses. *Communications of the Association for Information Systems*, vol. 53, no. 1, pp. 1-21, <https://doi.org/10.17705/1CAIS.05301>
- Venkatesh, V., Morris, M. G., Davis, G. B. & Davis, F. D., 2003. User Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly*, vol. 27, no. 3, pp. 425-478, <https://doi.org/10.2307/30036540>
- Wang, D., Churchill, E., Maes, P., Fan, X., Shneiderman, B., Shi, Y. & Wang, Q. (2020). From Human-Human Collaboration to Human-AI Collaboration, *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, <https://doi.org/10.1145/3334480.3381069>

- Williams, M. L., Saunderson, I. P. & Dhoest, A. (2021). Students' Perceptions of the Adoption and Use of Social Media in Academic Libraries: A UTAUT Study, *Communication*, vol. 47, no. 1, pp.76–94, <https://doi.org/10.1080/02500167.2021.1876123>
- Wirth, N., 2008. A Brief History of Software Engineering. *IEEE Annals of the History of Computing*, pp. 32-39.
- Wu, T., He, S., Liu, J., Sun, S., Liu, K., Han, Q.-L. & Tang, Y. (2023). A Brief Overview of ChatGPT: The History, Status Quo and Potential Future Development, *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 5, pp.1122–1136, <https://doi.org/10.1109/JAS.2023.123618>
- Yost, M. (2018). A Brief History of Software Development, Medium, Available Online: <https://medium.com/@micahyost/a-brief-history-of-software-development-f67a6e6ddae0>
- Zhang, B., Liang, P., Zhou, X., Ahmad, A. & Waseem, M. (2023). Practices and Challenges of Using GitHub Copilot: An Empirical Study, *International Conferences on Software Engineering and Knowledge Engineering*, <https://doi.org/10.18293/SEKE2023-077>
- Özpolat, Z., Yildirim, Ö. & Karabatak, M. (2023). Artificial Intelligence-Based Tools in Software Development Processes: Application of ChatGPT. *European Journal of Technique (EJT)*, vol. 13, no. 2, <https://doi.org/10.18293/SEKE2023-077>