

2016

Planerings Rapport

ALEXANDER PIHL & CHRISTIAN BENETT

HÖGSKOLAN I BORÅS

Contents

1.Introduction.....	2
2.Problem Discussion	4
3.Problemformulering.....	5
3.1 Formulering	5
3.2 Varför	5
Referenser	6

1.Introduction

Kan ett barn lära sig att programmera? Svaret är dock ganska självklart, med tanke på att programmering handlar om att lösa problem så behövs ett logiskt tänkande. Man skulle kunna göra en jämförelse med till exempel BRIOs tågbanor där man måste upprätthålla sig till vissa regler och begränsningar. Rälsbitarna sätter upp vilka begränsningar och regler vi måste handhålla oss till. På samma sätt fungerar det för en dator, där hårdvaran är begränsningen men där du och koden sätter reglerna.

Att barn har större och bredare tillgång till elektronik är ingen nyhet, själva användandet är vad som är den stora frågan. Surfplattan och mobilen har blivit en populär "barnpassare", detta är en forskningsfråga i sig vardera det är bra eller negativt. Genom lättare observationer om barns förmåga med att lära sig att bruka ny teknologi så är målet att forska på barns lärande och planeringsförmåga. Som (Pålsson S. 2016) skriver så är det inte lärandet av kodning som är i fokus. Barnet skall utveckla ett abstrakt, logiskt och analytiskt tänkande genom att arbeta systematiskt och på ett strukturerat sätt för att lösa problem. Att lära sig programmera kan vara allt ifrån svårt till lätt, ofta beror det till stor del på person till person men framförallt vad för språk och utvecklingsverktyg man arbetar med gentemot ens kunskap. Några av dom vanligaste språken är Java, C#, PHP, Python, C++, Assembler men det finns många fler, varje språk är "unikt" i sig men ändå väldigt lika. Språken delas upp i olika paradigmen för att ändå urskilja och gruppera efter liknelse.

Ett praktiskt experiment kommer att utföras mot/med barn för att kunna få en större klarhet och kunskapsbas angående just barn och programmering. Genom att utveckla en barnvänligt mobilapplikation där barn genom drag och släpp programmering, planera en rutt för en radiostyrd bil att ta sig igenom en förutbestämd bana. Barnet kommer på så sätt att skapa en algoritm vilket en programmera dagligen gör, en algoritm är en begränsad uppsättning väldefinierade instruktioner för att lösa en uppgift. Själva grund idén är att se om barn kan och vill lära sig att utveckla, genom att utforma detta till en lärande lek.

Barn och programmering är inget nytt och har därför med små steg blivit mer och mer populärt. Programmering handlar för det mesta enbart om ettor och nollor, alltså av och på men detta betyder inte att man skriver långa kodsnuttar med ettor och nollor. För detta har man alltså utvecklat utvecklingsmiljöer för att skriva koden, lika så finns det ofta en kompilator som omvandlar källkod till maskinkod som gör så att datorn kan processa datan. Källkoden är programmerarens grammatik, kallas även för syntax. Ett program skulle kunna jämföras med ett recept där man beskriver steg för steg om vad som skall göras.

Elektronik och teknologi har en explosionsartad utveckling och mycket har även utvecklats just för barn. Mycket av dagens barn elektronik är dock till större delen just för lek och underhållning medans en liten del är för utbildningssyfte.

Som (Johansson S. och Lindgren S. 2007 pp.9.) skriver är lek en viktig och en grundläggande betydelse för ett barns lärande och utveckling. Med detta sagt så anser vi att man måste hålla en bra balans mellan lek och lärandet.

I en studie som genomfördes på barn i Spanien (Natalia, David, Kieron & Carmen 2013) påvisas det att barn tycker det är mer givande att sitta med teknologiska apparater framför att sitta med penna och papper. Det genom att man har undersökt tillgången till Ipads och interaktive storys. Man har genom att ge barn tillgång till en Ipad och göra sina egna interaktiv storys. Slutsatsen från denna studie är att innehållet i en app och vad appen handlar om kan påverka värdet av barnets lärande när de använder sig av appar som medel för studier.

Det finns redan färdigutvecklade utvecklingsmiljöer för programmering för barn, några av dessa är Kojo, Scratch, Kids Ruby och Kodu alla dessa miljöer är utvecklade för att ett barn skall lätt kunna skapa något. Scratch har även en version som heter ScratchJr vilket målgruppen är barn 5-7 år och denna miljö är något lättare det vill säga en drag och släpp principen. Lego har utvecklat något som heter mindstorms och finns att köpa för allmänheten. Mindstorms riktar sig mot dock mot lite äldre barn tio år och uppåt. Principen är dock väldigt enkel där programmering fungerar på samma sätt som scratch där man drar och släpper olika kommandon för att få roboten att röra på sig.

Ett företag som använder sig utav mindstorms är (programming4kids 2016), detta är ett företag som riktar sig mot barn från ålder fem och uppåt. Dom menar att programmering kommer in väldigt sent i skolan om ens i huvudtaget, fördelningen mellan tjejer och killar är ojämn. Dem skriver att enligt EU kommissionen kommer det att saknas 1 miljon programmerare redan år 2020 och är något som präglar deras vision om programmering för barn, genom att erbjuda kurser och föreläsningar i skolor och där dom arbetar intensivt att få in programmering på schemat. "Vår vision är att alla svenska barn skall få möjligheten att lägga grunden för ett livslångt kreativt lärande med rötter i systemtänkande och naturvetenskap".

I (EU kommissionen 2014) pressmeddelande så uppmanar dom människor att delta i en europeisk kod vecka där dom påpekar hur viktigt det är med att lära sig kodning. Anledningen till dess viktighet beror på att mer än 90% av dagens yrken kräver någon kunskap inom informations- och kommunikationsteknik (IKT), där modern arbetskrafts förfrågan växer varje år med 3% vilket inte utexaminerade från datavetenskap håller samma takt.

2.Problem Discussion

Då barn blir mer och mer bekantade med programmering uppstår också frågan just hur man ska underlätta för att barn ska få den bästa möjliga miljön för att bli intresserad av programmering. Olika företag har lite olika lösningar just på hur man ska gå vidare för att barn ska få sig ett intresse för programmering. Några exempel är Lego Mindstorms eller appar liknande scratch. Det finns även studier som har som mål att undersöka hur barn lär sig av teknologiska apparater men alla studier som vi har läst har missat vad vi tycker är en viktig del inom lärandet av appar för barn. Det problemet som de mesta studier har missat är hur man kan använda just GUI för att påverka hur barnen lär sig.

Detta problem är att man oftast tittar på olika sätt att använda sig av grafiskt användargränssnitt (GUI) eller liknande för att barnet ska bli intresserad. Dock finns det lite eller inte någon forskning om hur designen på GUI kan påverka barnets sätt att lära sig. Kan till exempel GUI förenkla lärande av programmering hos barn eller minskar möjligheten att få ut något ur en app om GUI försvårar lärandet. Hur ska man då välja att designa GUI utifrån att man har barnets intresse för att lära sig i fokus? Den mer direkta frågan är väl för det första om man kan använda sig av avancerade teknologiska produkter för att få ett barn intresserad av programmering. Om det går att använda appar för att lära ut lära ut programmering till barn kommer det även att tillkomma frågor om hur gamla behöver barn vara för att lära sig av appar och hur ska själva appen se ut för att man ska underlätta lärandet så mycket som möjligt. Ska man exempel använda ett gränssnitt där man drar olika bitar för att bilda något, som Scratch eller lego mindstorms. Kanske är det bättre med fasta knappar för att barn ska få en så lätt förståelse som möjligt om hur allting fungerar. Det kan då sättas i kontrast till att man kommer att ha olika sätt att lära sig vilket kan gynnas av olika sätt att se på hur det GUI ska utvecklas.

Med tanke på att det kommer att fattas uppemot en miljon programmerare om tre år så är det en bra idé att lära sig utveckla redan i tidig ålder. Att kunna utveckla och förstå grundprinciperna som inte enbart handlar om att skriva kod. Utan den stora delen i utveckling handlar om logik, planering, analytiskt och systematiskt tänkande, vilket är bra områden att kunna.

3.Problemformulering

3.1 Formulering

Genom att studerat olika typer av utvecklingsmiljöer för barn som redan finns på marknaden, har vi kommit fram till att utveckla en mobilapplikation. Användaren skall planera med drag och släpp programmering, få en radiostyrd bil att röra sig i förutbestämd bana. Att använda sig utav en radiostyrd bil anser vi vara mer lockande för ett barn att “lek för att lära”, en fysisk artefakt som rör sig utefter vad barnet väljer skall locka mer.

Att det finns olika mobilplattformen det vet vi mycket väl, android, IOS, windows phone, symbian med flera är några av de många plattformarna. Att välja plattform att utveckla mot är någorlunda lätt då man oftast kommer ned till antingen android eller till IOS eller till båda. Att utveckla till android är både gratis och lätta att komma igång och även provköra sin applikation på en verklig fysisk plattform då det är ett java språk.

Skillnaden på android och apple är som jing och jang det vill säga dess motsatser, för att få utveckla mot apples plattform måste man ha ett konto som är kopplat som en utvecklare. Programvaran kostar även pengar för att kunna pröva på plattformar förutom datorns emulator. Att ens kunna använda utvecklingsmiljön så måste man installera detta på en mac det vill säga att man måste använda deras datorer för att ens kunna utveckla. Det finns möjlighet att utveckla från en PC men då måste man starta en virtuell maskin av mac för att då kunna utveckla.

Vi ansåg att utveckla mot IOS var ett stort plus då Sveriges kommuner använder sig utav apple produkter, men det skulle inte vara tidsmässig, ekonomiskt eller krånglighet mässigt möjligt. Vi valde att avskaffa denna idén och utveckla mot android istället.

3.2 Varför

Anledningen till att vi valde just att arbeta mot barn var på grund av utmaningen och att dem är våran framtid. Allt blir mer digitaliserat och automatiserat att man nästan inte behöver göra något själv nu för tiden. Om man kollar på dagens ungdomar, ställer man en lätt matematisk fråga så tar dom upp sin telefon för att sedan använda kalkylatorn. Detta är ett väldigt negativt beteende då hjärnan ej stimuleras. Att försöka få barn redan i tidig ålder att vilja lära sig något genom “lärande genom lek” skulle kunna vara ett sätt att lösa problemet.

Som sades i introduktionen så kommer det att saknas uppemot en miljon programmerare 2020, genom att introducera programmering tidigt i ett barns utveckling så kan man få upp intresset för IT.

Referenser

Europa Kommissionen (2014) *Skriv in i almanackan! Den europeiska kodveckan 11–17 oktober 2014. Gör dina idéer till verklighet med #coding*, pressmedelande 2014-06-04

Johansson, S. & Lindgren, S.(2007) *Lekens betydelse för barns lärande och utveckling: En studie av pedagogers och barns syn på fri lek i förskola och skola*, Kandidatuppsats ,Institutionen för pedagogik och didaktik samt Sociologiska institutionen: Göteborgs Universitet

Kucirkova,N. Messer,D . Panadero, C,F. Sheehy,K.(2013) *Children's engagement with educational iPad apps: Insights from a Spanish classroom* [2016-11-30]

Programing4kids.se (2016) *I Skolan* <http://programming4kids.se/programmering-i-skolan/> [2016-11-29]

Pålsson, S.(2016) *Forskning och erfarenheter kring programmering i skolan*
<http://omvarld.blogg.skolverket.se/2016/06/10/forskning-och-erfarenheter-kring-programmering-i-skolan/> [2016-11-28]