

Projektbeskrivning

Projektnamn	Travtipset
Projektägare	Mölk Mju23-1 - Ari
Projektdeltagare	Ari Ketola
Projektperiod	2024-05-16 - 2024-06-05

Sammanfattning

Projekt "Travtipset" av Ari Ketola.

Detta initiativ är en del av kursen Objektorienterad programmering 2 och syftar till att utveckla en applikation som genom maskininlärning förutser vinnarna i kommande travomgång. Applikationens mål är således att redovisa ett tipsförslag på vilka hästar som kommer att vinna kommande travlopp.

Med tidsbegränsningen från den 16:e maj till den 5:e juni, kommer projektet att använda en iterativ utvecklingsprocess med fokus på användarvänlighet och effektivitet. Med de få avgränsningar som projektuppdraget ger, kommer jag att prioritera kärnfunktioner i applikationen och anpassa utvecklingsstrategin baserat på feedback och tidsramar.

Beställare

Beställaren för "Travtipset" är Tomas Kindahl. Tomas Kindahl är en lärare i kursen Objektorienterad programmering 2 och det är inom ramen för denna kurs som projektet har initierats.

Syftet med att beställa detta projekt är flerfaldigt:

1. Pedagogiskt syfte: Att ge studenterna praktisk erfarenhet av objektorienterad programmering genom att utveckla en verklig applikation och erfarenhet i att skriva en projektspecifikation.
2. Utvärdering: Som en del av kursens bedömning vill beställaren se hur väl studenterna kan tillämpa de kunskaper de har förvärvat under kursens gång.

Beställaren förväntar sig att projektet ska redovisas både skriftligt och muntligt.

Bakgrund

Jag har ett hästintresse och även spelintresse. Att ta fram en applikation som genom maskininlärning förutspår sannolikheten att en specifik häst ska vinna ett travlopp förenar dessa två intressen med programmeringens fantastiska värld.

Förhoppningsvis ska även applikation generera framtida spelvinster.

Applikationens målgrupp blir alltså alla människor som vill spela på trav.

Syfte & mål

Syfte

Ta fram en applikation som förutspår vilka hästar som kommer att vinna ett specifik travomgång.

Mål

En applikation som ökar chansen att få alla rätt i en specifik travomgång. Applikationen ska på en tilltalande sätt redovisa ett förslag på travtips som tagits fram med hjälp av maskininlärning.

Metod / Modell

Användningen av en iterativ utvecklingsprocess där jag kontinuerligt går igenom design, utveckling, testning, dokumentation, use case-diagram och feedback-loopar.

Som en del av designfasen och för att bättre kommunicera och visualisera systemets arkitektur och komponenter, kommer jag att använda use case-diagram. Dessa diagram kommer att hjälpa mig att:

1. Definiera systemets komponenter och deras relationer.
2. Specificera beteendet hos olika delar av systemet.
3. Kommunicera designbeslut till externa intressenter.

Omfattning & strategi

Omfattning

Utveckling av en komplett applikation med grundläggande funktioner och ett tilltalande användargränssnitt.

Strategi

Genom att först fokusera på kärnfunktionalitet och sedan gradvis lägga till ytterligare funktioner baserat på feedback.

Avgränsningar

- Specifikationen skall leda till koden och koden skall implementera delar av specifikationen.
- Tidsramen för projektet är strikt begränsad till perioden från 16:e maj till den 5:e juni.
- På grund av den korta tidsramen kan vissa önskade funktioner eller förbättringar behöva skjutas upp till en eventuell framtida version av applikationen.

Tidsplan

Projektbeställarens milstolpar:

1. 16/5 Brainstorming.
2. 22/5 Ha bildat grupper.
3. 24/5 Grovskiss på projektspecifikation, inlämning på Moodle.
4. V22 Projektarbete.
5. 3/6 Avrundning, förberedelse för redovisning.
6. 5/6 Redovisning.

Organisation

Beställare: Tomas Kindahl (Mölk)

Projektdeltagare: Ari Ketola

Kravspecifikation

Projektkrav

- Grupper om tre personer.
- Börja med att skriva en projektspecifikation.
- Sedan implementera det man hinner med av projektspecifikationen.

Produktkrav

- En fungerande applikation utvecklad i valfritt programmeringsspråk med tillhörande dokumentation.

Produktdetaljer

- Python skript som skrapar ATG's hemsida och sparar ner data som ska användas till träningen av maskininlärningsmodellen. Detta skript körs bara en gång.
- Python skript som skrapar kommande travomgång från ATG.se hemsida och sparar ner data om denna travomgång.
- Python skript som tränar en maskininlärningsmodell och som sedan förutspår vinnarna i kommande travomgång med maskininlärningsmodellen.
- C# program som anropar ATG's API för att hämta kommande travlopp och sedan kör Python scripten. Programmet ska även ha inställningsmöjligheter för att ställa in erforderliga parametrar. Detta program ska slutligen redovisa vilka hästar som har störst sannolikhet att vinna de aktuella travloppen och eventuellt även fylla i denna data på ATG's hemsida.

Kommunikation & rapportering

Kommunikationsinsatser

Stämna av projektet med projektbeställaren senast 24:de maj. V22 möjlighet att delta i dagliga avstämningar 09:00 med projektbeställaren.

Extra möten vid behov. Möten cirka 15 minuter, eller så länge som krävs.

Använda Slack som kommunikationsverktyg, jag kommer även att använda Trello som projekthanteringsverktyg.

C# koden finns på <https://github.com/ariket/travtipset> .

Intressenter

Projektets beställare är Tomas Kindahl Molk, övriga intressenter är klass Mju23-1.

Rapportering & uppföljning

Skriftlig och muntlig redovisning, projektarbetet ska var färdigt till den 5 juni.

Projektet kommer att redovisas genom en inspelad video och inlämning av denna projektbeskrivning samt all tillhörande kod via git-hub.

Risker & kvalité

Riskanalys

Tekniska fel: Avbrott i ATG's API-tjänst kan påverka applikationens funktion. Åtgärd: implementera felhantering i koden för att informera användaren vid problem.

Tidsbegränsningar: Givet den strikta tidsramen för projektet finns risk att vissa funktioner inte hinner implementeras. Åtgärd: Prioritera kärnfunktionerna i applikation och se till att de är fullständigt fungerande innan ytterligare funktioner övervägs.

Otillräcklig testning: Brist på tid kan leda till otillräcklig testning, vilket kan resultera i buggar vid lansering. Åtgärd: Använda enhetstestning och betatestning i de tidiga skedena för att upptäcka och åtgärda buggar.

Sjukdom / Avhopp: Om projektmedlemmen blir sjuk kommer inte projektet att kunna slutföras. Åtgärd: Se till att inte bli sjuk.

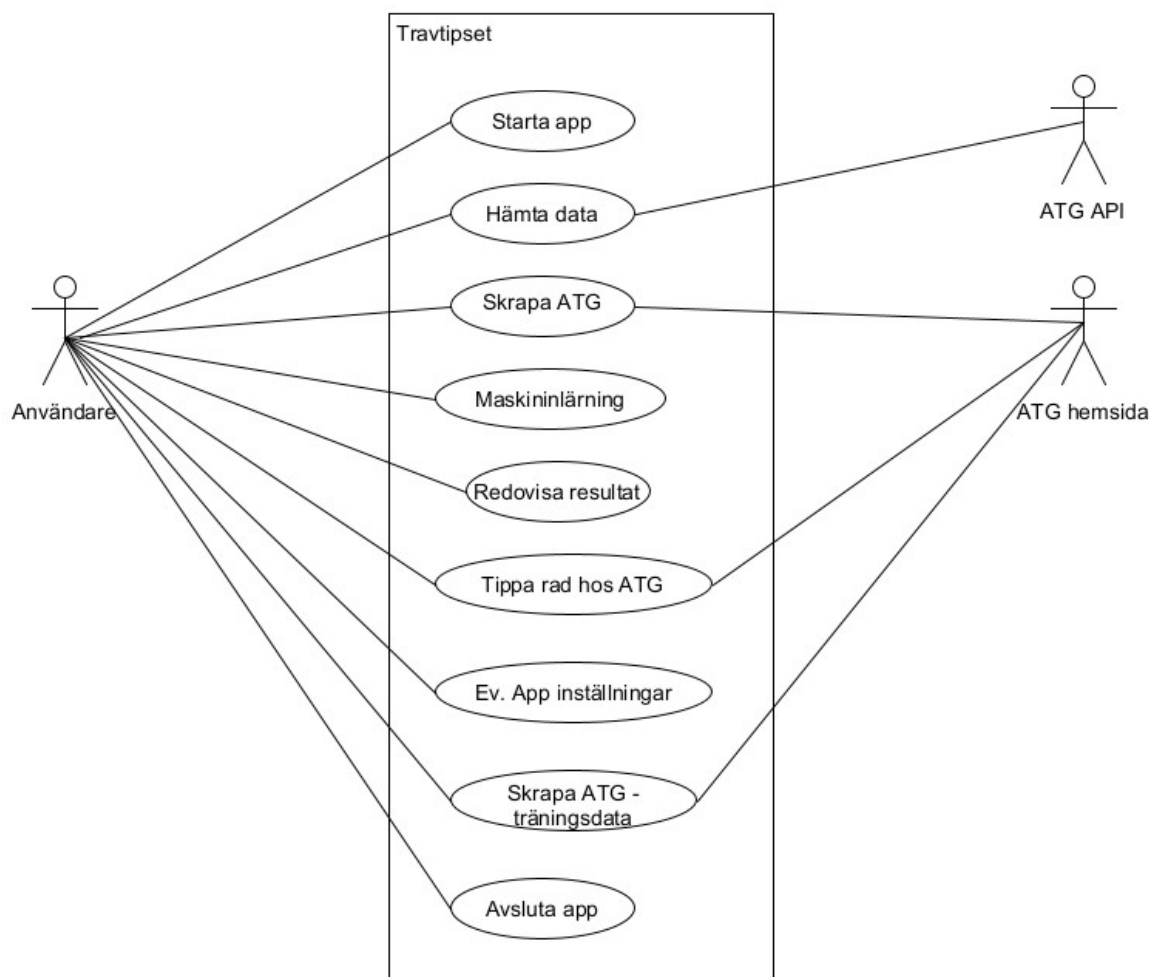
Risk	Sannolikhet	Konsekvens	Riskvärde	Riskåtgärd*
Fel på dator	2	4	8	Byt dator
Fel på Api	1	3	3	Använda sparad data
Otillräcklig test	2	3	6	Testa regelbundet
Tidsbegränsning	1	3	3	Hålla strikt tidsplan
Sjukdom	2	4	8	Övervaka arbetsbelastningen.

Kvalitetssäkring

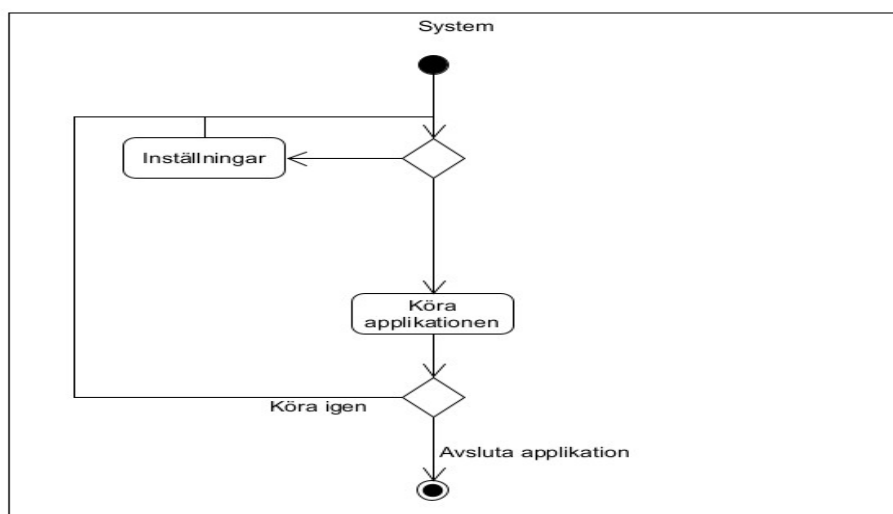
Kvalitetskontroller kommer att utföras efter varje utvecklingscykel, och tester kommer att utföras regelbundet.

UML

Use case diagram



Activity diagram



Use Case texter (listade i prioritetsordning)

- NAMN: Hämta data
- Aktörer: Användare, ATG's API
- Mål: Ladda hem data om nästa travomgång från ATG's API
- Förvillkor: Appen är startad.
- Trigger: Användaren går in i menyn och klickar på "Välj travlopp".
- Grundflöde:
 - 1.1. Användaren väljer speltyp.
 - 1.2. Appen kopplar upp mot ATG's API med vald speltyp.
 - 1.3. Appen laddar hem data om vald speltyp.
 - 1.4. Appen sparar nedladdad data i en fil.

Kommentarer: Möjliga speltyper att välja är: V64,V75 eller V86.

- NAMN: Skrapa ATG
- Aktörer: Användare, ATG's hemsida
- Mål: Skrapa hem detaljerad data om nästa travomgång från ATG's hemsida
- Förvillkor: Data om nästa travomgång(Use Case "Hämta data") är hämtad
- Trigger: Användaren går in i menyn och klickar på "Hämta travlopp".
- Grundflöde:
 - 1.1. Appen anropar ett Python skript som genomför skrapningen.
 - 1.2. Python skriptet skrapar data om travloppen för angiven travomgång.
 - 1.3. Python skriptet sorterar skrapad data.
 - 1.4. Python skriptet sparar skrapad och sorterad data i en fil.

Kommentarer: Möjliga speltyper är: V64,V75 eller V86.

- NAMN: Maskininlärning
- Aktörer: Användare
- Mål: Förutse vilka hästar som har störst chans att vinna i travloppen
- Förvillkor: Bägge Python scripten som skrapar ATG måste vara avslutade
- Trigger: Use Caset "Skrapa ATG" är genomförd
- Grundflöde:
 - 1.1. Appen anropar ett Python skript som genomför maskininlärningen.
 - 1.2. Python skriptet tränar modellen med träningsdata.
 - 1.3. Python skriptet förutspår m.h.a. modellen vilka hästar som har störst sannolikhet att vinna travloppen i angiven travomgång.
 - 1.4. Python skriptet sparar resultatet i en fil.

- NAMN: Redovisa resultat
 - Aktörer: Användare
 - Mål: Redovisa vilka hästar som har störst sannolikhet att vinna i travomgången
 - Förvillkor: Python skriptet som kör maskininläringen måste vara avslutad
 - Trigger: Use Case "Maskininläring" är genomförd
 - Grundflöde:
 - 1.1. Appen hämtar resultatet från filen som Python skriptet sparar.
 - 1.2. Appen redovisar resultatet på skärmen till användaren.
-
- NAMN: App inställningar ----->"framtida utveckling"
 - Aktörer: Användare
 - Mål: Ge användaren möjlighet att ändra erforderliga parametrar som krävs för att köra appen
 - Förvillkor: Appen är startad
 - Trigger: Användaren går in i menyn och klickar på "Inställningar"
 - Grundflöde:
 - 1.1. Appen visar vilka inställningar som kan ändras.
 - 1.2. Användaren väljer inställning.
 - 1.3. Användaren ändrar inställningen.
 - 1.4. Appen kontrollerar att ifylld inställning är korrekt ifylld.
 - 1.5. Appen sparar ändrade inställningar.
-
- NAMN: Tippa rad hos ATG ----->"framtida utveckling"
 - Aktörer: Användare, ATG's hemsida
 - Mål: Skicka resultatet till ATG's hemsida och fyll i hemsidan på ett korrekt sätt
 - Förvillkor: Python skriptet som kör maskininläringen måste vara avslutad
 - Trigger: Användaren går in i menyn och klickar på "Skicka resultat"
 - Grundflöde:
 - 1.1. Appen anropar ett Python skript.
 - 1.2. Python skriptet fyller i resultatdatan på ATG.se travkupon.
-
- NAMN: Skrapa ATG - träningsdata
 - Aktörer: Användare, ATG's hemsida
 - Mål: Skrapa hem detaljerad data om äldre travomgångar från ATG's hemsida
 - Förvillkor: Ange vilka travomgångar som efterfrågas
 - Trigger: Efterfrågade travomgångar har angetts
 - Grundflöde:
 - 1.1. Python skriptet skrapar data för travloppen i angivna travomgångar.
 - 1.2. Python skriptet sorterar skrapad data.
 - 1.3. Python skriptet sparar skrapad och sorterad data i en fil
- Kommentarer: Denna skrapningen är redan genomförd innan projektstart