

# DD1350 Logik för dataloger

## Laboration 2

Erik Ringdahl, erikrin@kth.se  
Joel Tjärnstig, joelt@kth.se

11 oktober 2015

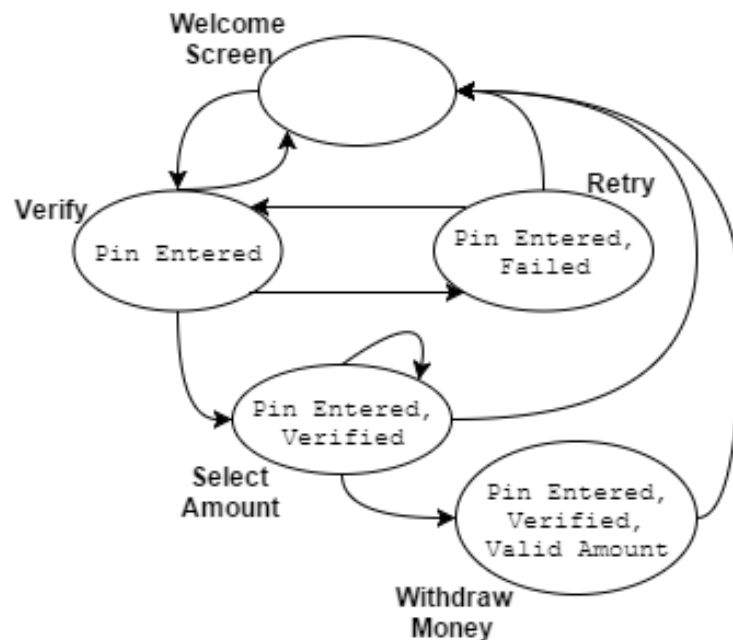
### **1 Verktögsutveckling**

## 2 Modellering

Vi valde att modellera en förenklad uttagsautomat. En användare möts först av en välkomstkärm. Användaren verifieras genom att ange sin pinkod. Om pinkoden är fel, bes användaren ange den igen. Om användaren anger rätt pinkod får ett belopp anges. Om beloppet inte kan tas ut får användaren skriva ett nytt belopp. Om beloppet finns tillgängligt tas pengarna ut och uttagsautomaten återgår till välkomstkärmen. Användaren kan även närsomhelst avbryta uttaget och återgå till välkomstkärmen.

### 2.1 Tillståndsgraf

Vi har 5 olika tillstånd (**Welcome Screen**, **Verify**, **Retry**, **Select Amount**, **Withdraw Money**) och 4 atomer (Pin Entered, Failed, Verified, Valid Amount).



Figur 1: Tillståndsgraf för uttagsautomat

## 2.2 Prolog-kompatibel representation

```
% States are ws(welcome screen), ve(verify), re(retry),  
% sa(select amount) and wm(withdraw money).  
[  
  [ws, [ve]],  
  [ve, [ws, ve, re, sa]],  
  [re, [ws, ve]],  
  [sa, [ws, sa, wm]],  
  [wm, [ws]]  
].  
  
% Labeling  
[  
  [ws, []],  
  [ve, [pe]],  
  [re, [pe, f]],  
  [sa, [pe, v]],  
  [wm, [pe, v, va]]  
].
```

## 3 Specifiering

### 3.1 Hållbar systemegenskap

Det finns en stig där så småningom det går att ta ut pengar.

$\text{ef}(\text{and}(\text{and}(\text{pe}, \text{v}), \text{va}))$

### 3.2 Ohållbar systemegenskap

Det finns en stig där något av nästa steg inte är välkomstskärmen.

$\text{ef}(\text{not}(\text{ex}(\text{not}(\text{pe}))))$

## 4 Verifiering

### 4.1 Systemegenskaper

### 4.2 Tester

Modellproveraren klarade alla tester.

## Appendix

### A Källkod