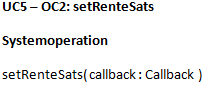
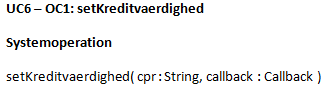
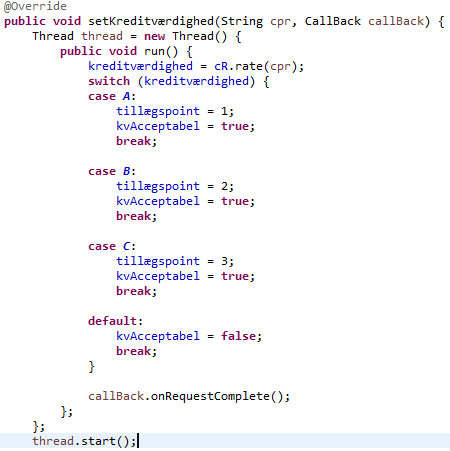
## Operationskontrakter

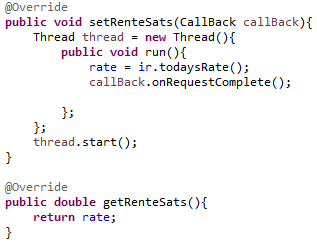
 

Ved operationskontrakterne har vi valgt og fokusere på centrale dele af systemet. Ved Usecase diagrammet blev der vist, en tydelig forbindelse imellem UC5+UC6 til UC7. I eksemplet ovenover ser vi vores første operationskontrakt: setKreditvaerdighed. Hvis vi ser på parameterne på denne metode(og får OC2), er der klart at en parameter springer i øjene: Callback.

Ideen med Callback parameteren var, at vi allerede her vist at disse to metoder(setKreditvaerdighed, setRenteSats) skulle køres i tråde. Her ser i koden til setKreditvaerdighed:

  
Det callback parameteren gør, er at det tillader os og kalde en metode som hedder onRequestComplete();

Denne metode er en public abstract klasse i Callback klassen. Som også er abstract. Og hvad den egentlig gør at efter tråden er blevet startet, bliver der oprettet forbindelse til RKI, hvilket så resultere i en kreditværdighed. Dette bliver så tjekket igennem switch sætningen, hvor efter callback kalder onRequestComplete(); på sig selv. Dette gør at systemet ved præcis hvornår metoden setKreditværdighed() er færdig. Ellers skulle systemet sidder og pinge i et fast interval, for og finde ud af hvornår den er færdig.

  
Ovenfor ser vi metoden setRenteSats(); Ideen er den samme som for setKrediværdighed(). Dette gør vores 2 tråde meget mere effektive, eftersom at systemet kan fokuser på andre opgaver, indtil at onRequestComplete(); bliver kaldt.

Operationkontrakterne kan ses i deres helhed i bilagene.