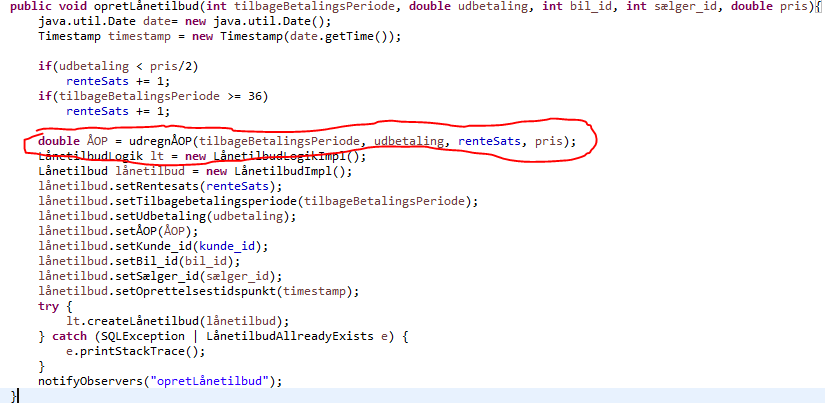
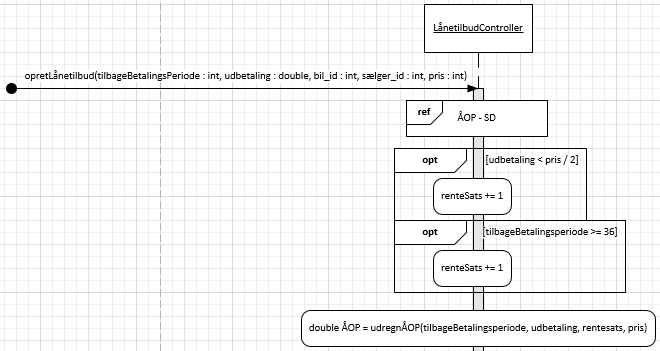
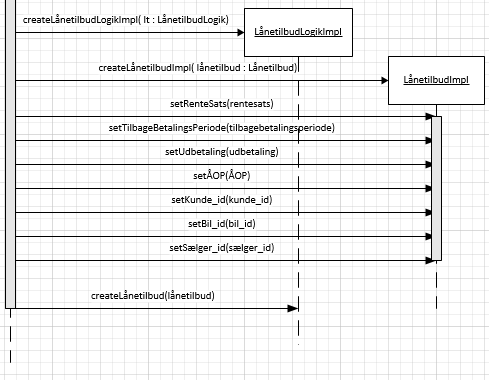


Sekvensdiagrammet for udregnÅOP viser primært hvordan vi kommer frem til ÅOP. Der bliver sat en masse instanser i starten til en masser forskellige værdier. Men det interresante sker i loopet. I loopet finder vi summen, som vi bruger til og udregne OP(Omkostnings procent). Dette gør vi ved kører en for løkke i gennem, for og finde frem til summen. For løkken kører en gang for hver måned tilbageBetalingsPeriode er blevet sat til. Vi ender så ud med og få OP, som vi skal bruge i formlen til ÅOP. LånetilbudController retunerer så tilsidst ÅOP til den der har kaldt metoden. 

I dette eksempel kan vi se(markeret med rødt) at udregnÅOP bliver kald når et lånetilbud skal oprettes. Det er også vigtig og bemærke at, rentesatsen der bliver parameter i udregnÅOP, er den daglige rentesats fra banken. Denne rentesats bliver nemlig altid beregnet, før man kalder opretlånetilbud. Dette kan ses i sekvensdiagrammet for beregnLånetilbud.



Sekvensdiagrammet for opretLånetilbuddet, ser vi den tydelige ref, vi nævte før. Går vi længere ned af livslinjen ser vi også at vi tager højde for at hvis udbetalingen er mindre en 50% af den absolute pris, så skal der ligges +1 procentpoint til rentesatsen. Dette gælder også hvis at tilbageBetalingsperioden er mere en 35 måneder. Efter vi så har den rigtige rentesats, bliver udregnÅOP kaldt.



Senere på livslinjen bliver der lavet 2 objekter: LånetilbudLogikImpl og LånetilbudImpl. Herefter bliver alle de nødvendige instanser for og lave et låetilbud sat på LånetilbudImpl. Efter dette bliver der så sendt et lånetilbud afsted til LånetilbudLogikImpl, som så ligger lånetilbuddet ind i databasen. Sekvensdiagrammet for opretLånetilbud kan ses i dets helhed i bilagene.