## Administrator

Context administrator::create account(): void

Pre: self.Username != User.username

Post: self.CreateUser

-- Je kan alleen een account maken voor iemand die er nog geen heeft.

Context administrator inv:

Asset.isactive&&isownedbyCliënt

--Cliënts kunnen enkel voor al hun actieve assets de support verlengen.

## Handler

Context Handler inv:  
self.Children->Count() > 0 implies self.Children->forAll(c | c.parent = self)  
*-- Als het ticket child-tickets heeft, moet de parent van ieder child ticket het ticket zelf zijn*

Context Handler::assignToHandler(): void  
pre: self.handledBy->isEmpty()  
post: self.handledBy->notEmpty()  
*-- Een ticket toewijzen aan een behandelaar kan enkel als het nog niet is toegewezen*

Context Handler::appendChildTicket(child: Ticket):void  
pre: child.Parent->isEmpty()  
post: self.children->includes(child) and child.parent->includes(self)  
*-- Indien een child ticket wordt toegekend aan een ticket, mag het toegewezen ticket nog niet tot een andere parent behoren. Na de actie zal het parent-ticket naar het child-ticket verwijzen en zal het child-ticket tot het parent-ticket behoren*

Context Handler::appendToParentTicket(parent: Ticket): void  
pre: self.Parent->isEmpty()  
post: self.parent->includes(parent) and parent.children->includes(self)  
*-- Indien een ticket aan een parent wordt toegekend, mag het zelf nog niet aan een andere parent zijn toegekend. Na de actie zal het ticket naar het parent-ticket verwijzen en zal de parent naar het child-ticket verwijzen*

Context Handler::relateToAsset(asset: Asset): void  
pre: self.relatedAssets->excludes(asset)  
post: self.relatedAssets->includes(asset)  
*-- Indien een asset aan een ticket wordt gekoppeld, mag het nog niet aan dat ticket zijn gekoppeld.*

Context Handler :: changeImpact(newImpact:enumImpact):void

Pre: Ticket.impact->isEmpty()

Post: ticket.impact->isnotempty

*--Als de Impact nog niet ingesteld is moet dit 1e gebeuren. Kan daarna nog gewijzigd worden.*

Context Handler :: changeUrgency(newUrgency:enumUrgency)

Pre: Ticket.urgency->isEmpty()

Post: ticket. urgency ->isnotempty()

*--Als de urgency nog niet ingesteld is moet dit 1e gebeuren. Kan daarna nog gewijzigd worden.*

## User

Context User::changeStatus(): void  
pre: self.status = not (enumStatus.Resolved or enumStatus.Rejected or enumStatus.Failed)  
*-- Een ticket dat een van de afgesloten statussen heeft, kan niet meer van status worden veranderd*

*-- Als de Account en het wachtwoord correct zijn ga je naar de interface van die gebruiker.*

*-- Wachtwoord moet aan de criteria voldoen.*

*-- Als de user een handler/admin is kan een bericht ook privé zijn. Anders altijd public.*

## Cliënt Contact

--Kan enkel zijn eigen Cliënt tickets/berichten zien.