1.4.1 Vergwleich von queue und stack  
Bei einer Queue wird das Element, das am längsten in der Queue ist, als erstes entfernt. Bei einem Stack wird das Element, das zuletzt hinzugefügt wurde, als erstes entfernt.  
Bei einer einfach verketteten Liste handelt es sich um eine Liste, bei der jeder Knoten auf den nächsten Knoten in der Liste verweist. Durch eine Iteration über die Liste kann auf jeden Knoten zugegriffen werden  
und zum Beispiel eine Löschung oder eine Einfügung durchgeführt werden.  
Bei einer einfachverketteten Liste muss beim Einfügen geschaut werden ob es schon einen Knoten gibt, wenn nein muss einer angelegt werden. Wenn ja muss man zwischen drei Einfügeoptionen entscheiden. Die Liste muss durchgelaufen werden bis das nextKnoten Attribut  
der Liste null ist und der Knoten dahinter eingefügt werden kann (einfügen am Ende). Das erste Element muss zwischengespeichert werden, und das neue Element wird bekommt als nextKnoten das zwischengespeicherte Element (Einfügen am Anfang).  
Die liste muss bis zur gewünschten Stelle durchlaufen werden und das Element davor und danach müssen umgebogen werden, sodass das neue Element dazwischen eingefügt werden kann (Einfügen in der Mitte).  
Beim Löschen muss das ganze genau andersrum gemacht werden.  
Bei der Queue und einem Stack gibt es nur das am Anfang und am Ende einfügen und löschen.  
Default Opperatoren:  
Linkedlist: hat keine direkte Javaimplementation, sondern muss selbst gecoded werden.  
Queue: offer, poll, peek oder add, remove, element  
Stack: push, pop, peek oder add, remove, element  
Die zweite Option gibt ein fahler auf wenn es zu einem Fehler kommt, die erste Option gibt false zurück.  
  
1.5.1  
Beim Anwendungsfalldiagramm handelt es sich um ein Diagramm, welches die Interaktion zwischen einem System und einem Akteur darstellt. Dabei wird der zeitliche Ablauf der Interaktionen dargestellt.  
Es gibt Akteure in Form von Strichmännchen, Anwendungsfälle