Anforderungen für CPV

## Auftraggeber: HAE

## Projektverantwortliche: EG

## Betreuer: MMU, TB

## System

Entwicklung Visualisierungstoll Crane Position Visualizer „CPV“

Eine Unterstützung für die SPS Software Entwicklung bei der Firma Künz entwickelt werden. Folgenden Aufgaben sollen unterstützt:

* Die dynamische uns schematische Darstellung von
* Liniensegment: lineSeg()
* Rechteck: rectangle()
* Kapsel: capsule()
* Kreis: circle()
* Ziel: target()

## Zweck

Zweck dieses Dokuments ist die Beschreibung und genaue Definition der Basisanforderungen an die zu realisierende Software.

Zweck von der Tool ist die visuelle Unterstützung der Kranentwicklung mit SPS.

## Umfang

Dieses Dokument umfasst eine textuelle Beschreibung und graphische Veranschaulichung der Anforderungen an das Visualisierungstool, und somit zu realisierende Teile der Software innerhalb des Praktikums.

## Referenzen

Projektbeschreibung von Hannes am 12.07.2017

Besprechung mit Markus und Hannes am 17.07.2017

Feedback von Hannes am 27.07.2017

Besprechung mit Martin und Markus am 01.08.2017

Besprechung mit MMU, HAE und TB 14.09. 2017

## Funktionale Anforderungen

## REQ1 Hintergrund

Darstellungen eines Hintergrundes, an welche die statische Komponente visualisiert werden .png .jpeg

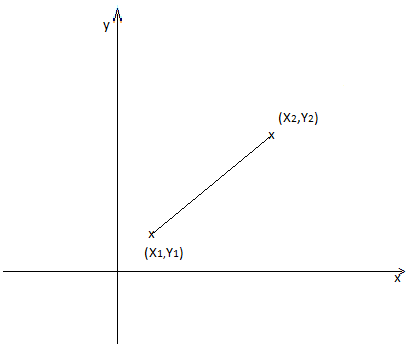
## REQ2 Koordinaten

Die Punkte sind in kartesischen Koordinatensystem definiert. Die Größe des Workspaces wird im Program definiert.

## REQ3 Liniensegment

Funktion: lineSegment(x1 , y1, x2 , y2 , color)

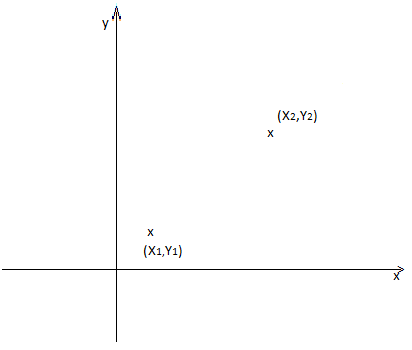
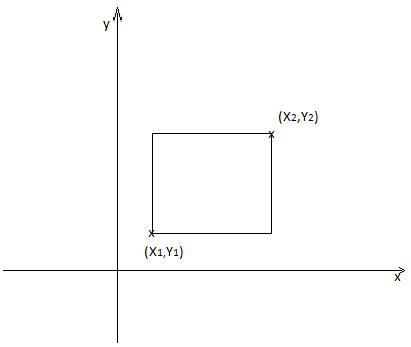
Ein Liniensegment ist durch vier Koordinaten definiert, und kann verschiedenen Farben haben.



## REQ4 Rechteck

Funktion: rectangle(x1 , y1, x2 , y2 , color)

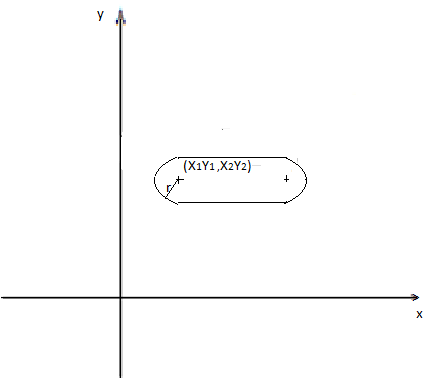
Ein Rechteck wird durch die links unteren zwei Koordinaten und die rechts oberen zwei Koordinaten definiert, und kann verschiedenen Farben haben.

## REQ5 Kapsel

Funktion: capsule(x1 , y, x2 , y , r, color)

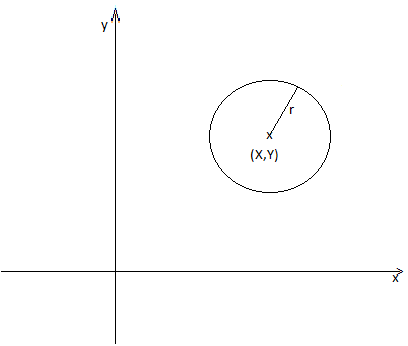
Eine Kapsel wird durch die vier Koordinaten des Brennpunktes und ein Radius definiert, und kann verschiedenen Farben haben.



## REQ6 Kreis

Funktion: circle(x , y , r , color)

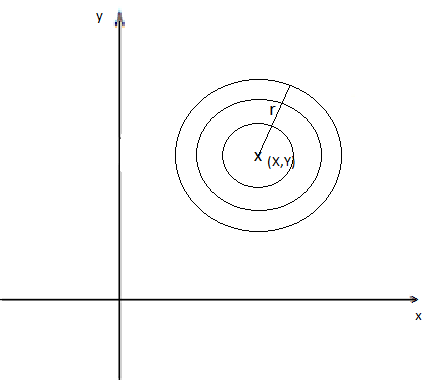
Ein Kreis wird durch die Mittelpunktkoordinate und ein Radius definiert, und kann verschiedenen Farben haben.



## REQ7 Ziel

Funktion: target(x , y , r , color)

Target ist eine Sonderform des Kreises, weil es sich hier um konzentrische Kreise handelt.



## REQ8 SPS Schnittstelle

## SPS Schnittstelle wird wie gefolgt definiert

TargetArray[]

X Dint

Y Dint

R Dint

RectangleArray[ ]

X1 Dint

Y1 Dint

X2 Dint

Y2 Dint

CircleArray[ ]

X Dint

Y Dint

R Dint

CapsuleArray[ ]

X1 Dint

Y1 Dint

X2 Dint

Y2 Dint

R Dint

LineArray[ ]

X1 Dint

Y1 Dint

X2 Dint

Y2 Dint

## REQ9 Zoom

Es soll möglich sein mit dem Mausrad -an der Position von der Maus- in die Visualisierung hinein oder hinaus zu zoomen. Das Bild soll mit scaliert werden.

## REQ10 Bewegen, Verschieben

Die Visualisierung soll auch bewegbar sein. Es soll möglich sein einen nicht sichtbaren Bereich wieder in sichtbaren Bereich zu holen.

## REQ11 Leertaste Funktion

## Es soll bei der Betätigung der Leertaste wieder die ganze Visualisierung an Bildschirm sichtbar sein, mit allen dargestellten Elementen, wie bei dem Start der Anwendung.

**Anwendungsbeispiel:**

**Bahnplanung:**

## Das Hintergrundbild stellt die Workspace, Gelände dar.

## Die Hindernisse, Sperr- oder Sicherheitszonen sollen als Rechtecke gegeben falls als Kreise dargestellt werden

## Die Katze soll ebenfalls als Rechteck visualisiert werden.

## Das Haupträger des Kranes und den Geschwindigkeitsvektor von der Krankatze zu Ziel sind als Liniensegment darzustellen.

## Die Kapseln symbolisieren jeweils den Bremswegen.