Name: David Mayr

Email: david.mayr20@gmail.com

**1.1 Visionen und Kurzbeschreibung des Projekts:**

Projektname: **Planetensystem**

Projektbeschreibung: Berechnung der Planetenbahnen rund um die Sonne und relativen Stand zueinander, abhängig vom Datum ( ev. noch die Mondbahnen und Mondkonstellationen 🡪 Jupiter, Saturn ).

Darstellung des Sonnensystems durch Speichern in einer Datei und über Browser ausgeben, oder grafisch als 2D-Modell od. 3D-Modell.

Ziel: es soll eine Echtzeitdarstellung des Sonnensystems im einem C++-Algorithmus realisiert werden, wodurch der genaue Stand der Planeten in Relation zur Sonne und zueinander je nach eingegebenem Datum dargestellt werden kann.

Plattform: Visual Studio 2017

**2.1 Zusammenhang mit bereits bestehenden Systemen:**

Das Sonnensystem wird definiert als der Bereich im Universum, der sich von der Sonne bis hin zum

äußersten Planeten des Systems erstreckt, in unserem Fall ist das Pluto.

Der Hypothese nach existiert darüber hinaus eine Oorthsche Wolke bestehend aus einer

Ansammlung astronomische Objekte (Kometen), die sich wie eine Blase rund um das Sonnensystem

schließt. Da es sich hierbei jedoch bisher nur um eine Hypothese handelt, wird diese Wolke nicht ins

System miteinbezogen.

Das Modell befasst sich ausschließlich mit den Daten der Sonne und den sie umkreisenden Planeten

(ev. werden noch Teile des Asteroidengürtels miteinbezogen)

Des weiteren wird die Entfernung der einzelnen Himmelskörper zur Sonne und zueinander je nach

festgelegter Zeit berechnet.

**2.2 Überblick über die geforderte Funktionalität:**

Berechnung der Planetenbahnen:

Jeder Planet, der um die Sonne rotiert folgt einer bestimmten

Bahn, die sich aus seiner Masse, der Umlaufgeschwindigkeit und der Entfernung zur Sonne definiert.

Diese Bahn soll anhand der Daten des Planeten (Masse, Umlaufzeit, ...) berechnet werden.

Daten zu den Planeten und zur Sonne:

zu jedem Himmelskörper (einschließlich zur Sonne) sollen Eckdaten, wie der Name, die Größe (in

Volumen u. Durchmesser), die Masse, Eigenrotation (Dia Solar, Dia Sid.), Anzahl der Monde (und

deren Namen), Entfernung zur Sonne und die Dauer einer Umrundung (Jahr) bekanntgegeben

werden.

Die Konstellation der Planeten sollte in Echtzeit berechnet werden, d.h. die Bewegung rund um die

Sonne bei Verstreichen der Zeit nachgestellt werden.

Der Stand der Planeten in Relation zueinander und zur Sonne soll daher zu jedem Zeitpunkt bekannt

sein.

Wenn möglich sollte eine 2D- od. 3D-Darstellung des Systems angezeigt werden, welches anhand der errechneten Daten ein Modell der Lage ausgibt.

**3. Detaillierte Spezifikation:**

Anforderungen: der Benutzer soll über ein Menü einen Planeten auswählen können und

dann die Daten dieses Planeten angezeigt bekommen. Es soll auch ein Zeitraum eingegeben werden, wonach die sich veränderte Position und Entfernung des Planeten zur Sonne angezeigt wird, sowie die Strecke, die der Planet inzwischen zurückgelegt hat.

Schnittstellen: Texteingabe über Konsole via Menü (siehe Menüplan)

od. graphisch über Panel mit QT. (optional)

Bei graphischer Oberfläche kann optional auch noch eine 2D (3D) – Ansicht für die momentane Planetenkonstellation angezeigt werden.

Die Daten für die Planeten, sowie deren Berechnungen für den jeweiligen Stand auf der Umlaufbahn werden in einer CSV-Datei gespeichert, von wo aus sie für weitere Anwendungen od. Abfragen abgerufen werden können.

**Skizze Menüplan:**

>> Bitte den Namen des Planeten eingeben (od. Programm beenden mit EXIT) :

<< *NamePlanet*

>> Name: (*Bsp.: Saturn*)

>> Größe: *Durchmesser, Volumen*

>> Masse: *kg*

>> Anzahl der Monde:

>> Eigenrotation: *Winkelgeschw. (rad/sec.), Tag (Dia Solar, Dia Sid.)*

Datum: *momentanes Datum*

>> Entfernung zur Sonne: *km, AE*

>> Umlaufzeit um die Sonne: *Tage (Jahre)*

>> Bitte einen Zeitraum eingeben, wonach der neue Stand berechnet werden soll

(od. Programm beenden mit EXIT):

<< *Zeitraum (in Tagen, Jahren)*

>> die zurückgelegte Bahnstrecke (Umrundungen): *km, Umrundungen*

>> derzeitige Entfernung zur Sonne

1. >> Abfrage: *Soll die Entfernung zu einem weiteren Planeten bestimmt werden?*

*<< ja* --> \*)

*<< nein:* Programm kehrt zum Ursprung zurück.

\*) >> Name des Planeten:

<< *Name*

>> Die Entfernung zum ausgewählten Planeten beträgt: *Strecke in km, AE*

>> *1)*