ENTWURFSDOKUMENT

(V. 1.0)

KNOT^3

PSE WS 2013/14

Auftraggeber:
Karlsruher Institut für Technologie
Institut für Betriebs- und Dialogsysteme
Prof. Dr.-Ing. C. Dachsbacher

Betreuer:
Dipl.-Inf. Thorsten Schmidt
Dipl.-Inf. M. Retzlaff

Auftragnehmer:
Tobias Schulz, Maximilian Reuter, Pascal Knodel,
Gerd Augsburg, Christina Erler, Daniel Warzel

15. Dezember 2013

Inhaltsverzeichnis

Kapitel 1

Einleitung

Das Knobel- und Konstruktionsspiel Knot[3], welches im Auftrag des IBDS Dachsbacher ausgearbeitet wird, wird wie im Pflichtenheft spezifiziert angefertigt.

Kapitel 2

Aufbau

2.1 Architektur

Die grundlegende Architektur des Spiels basiert auf der Spielkomponenten-Infrastruktur des XNA-Framework, die mit Spielzuständen kombiniert wird. Die abstrakten Klassen GameStateComponent und DrawableGameStateComponent erben von den von XNA bereitgestellten Klassen GameComponent und DrawableGameComponent implementieren zusätzlich die Schnittstelle IGameStateComponent. Sie unterscheiden sich von den XNA-Basisklassen dadurch, dass sie immer eine Referenz auf einen bestimmten Spielzustand halten und nur in Kombination mit diesem zu verwenden sind.

Die Spielzustände erben von der abstrakten Basisklasse GameScreen und halten eine Liste von IGameStateComponent-Objekten. Wird ein Spielzustand aktiviert, indem von einem anderen Spielzustand aus zu ihm gewechselt wird oder indem er der Startzustand ist, dann weist er seine Liste von IGameStateComponent-Objekten dem Components-Attribut der Game-Klasse zu, die von der vom XNA-Framework bereitgestellten abstrakten Klasse Game erbt. So ist zu jedem Zeitpunkt während der Laufzeit des Spiels ein Spielzustand aktiv, der die aktuelle Liste von Spielkomponenten verwaltet.

Die Spielkomponenten, die nicht gezeichnet werden und nur auf Eingaben reagieren, haben nur eine Update()-Methode und erben von GameStateComponent. Dies sind vor allem verschiedene Input-Handler, welche Tastatur- und Mauseingaben verarbeiten und beispielsweise die Kameraposition und das Kameratarget ändern oder Spielobjekte bewegen.

Spielkomponenten, die neben der Update()-Methode auch eine Draw()-Methode besitzen, erben von DrawableGameStateComponent. Dies sind vor allem die Elemente, aus denen die grafische Benutzeroberfläche zusammengesetzt ist, deren abstrakte Basisklasse Widget darstellt. [weitere Erklärungen zu Widgets...]

Alle Spielobjekte implementieren die Schnittstelle IGameObject. Die abstrakte Klasse GameModel repräsentiert dabei ein Spielobjekt, das aus einem 3D-Modell besteht, und hält zu diesem Zweck eine Referenz auf ein Objekt der Klasse Model aus dem XNA-Framework sowie weitere Eigenschaften wie Position, Drehung und Skalierung.

Spielobjekte sind keine Komponenten, sondern werden in einer Spielwelt zusammenfasst, die durch die Klasse World repräsentiert wird. Die Spielwelt ist ein DrawableGameStateComponent und ruft in ihrer Update()- und Draw()-Methoden jeweils die dazugehörigen Methoden aller in ihr enthaltenen Spielobjekte auf.

Shadereffekte werden durch die abstrakte Klasse RenderEffect und die von ihr abgeleiteten Klassen gekapselt. Ein RenderEffect enthält ein Rendertarget vom Typ RenderTarget2D als Attribut und implementiert jeweils eine Begin()- und eine End-Methode. In der Methode Begin() wird das aktuell von XNA genutzte

Rendertarget auf einem Stack gesichert und das Rendertarget des Effekts wird als aktuelles Rendertarget gesetzt.

Nach dem Aufruf von Begin() werden alle Draw()-Calls von XNA auf dem gesetzten Rendertarget ausgeführt. Es wird also in eine im RenderTarget2D-Objekt enthaltene Bitmap gezeichnet. Dabei wird von den Draw()-Methoden der GameModels die DrawModel(GameModel)-Methode des RenderEffects aufgerufen, der die Modelle mit bestimmten Shadereffekten in die Bitmap zeichnet.

In der End()-Methode wird schließlich das auf dem Stack gesicherte vorher genutzte Rendertarget wiederhergestellt und das Rendertarget des RenderEffects wird, unter Umständen verändert durch Post-Processing-Effekte, auf dieses übergeordnete Rendertarget gezeichnet.

2.2 Klassendiagramm

2.3 Verwendete Entwurfsmuster

Kapitel 3

Klassenübersicht

Kapitel 4

Abläufe

4.1 Sequenzdiagramme

Kapitel 5

Klassenindex

5.1 Klassen

5.1.1 Klasse ArrowModel

Beschreibung:

test

Eigenschaften:

public ArrowModelInfo Info

Methoden:

public void Draw (GameTime)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public void ArrowModel (GameScreenArrowModelInfo)

public void Update (GameTime)

5.1.2 Klasse ArrowModelInfo

Beschreibung:

Eigenschaften:

public Vector3 Direction

Methoden:

public void ArrowModelInfo (Vector3Vector3)

5.1.3 Klasse AudioSettingsScreen

Beschreibung:

Eigenschaften:

protected void SettingsMenuSection

Methoden:

public void Update ()

5.1.4 Klasse Camera

Beschreibung:

Eigenschaften:

private void World

public Vector3 Position

public Vector3 Target

public float FoV

public Matrix ViewMatrix

public Matrix WorldMatrix

public Matrix ProjectionMatrix

public Vector3 ArcballTarget

public BoundingFrustum ViewFrustum

Methoden:

public Vector3 TargetDirection (Vector3)

public float TargetDistance (float)

public void Camera (GameScreenWorld)

public Ray GetMouseRay (Vector2Ray)

public void Update (GameTime)

5.1.5 Klasse CelShadingEffect

Beschreibung:

Methoden:

protected void DrawRenderTarget (GameTime)

public void DrawModel (GameModelGameTime)

public void RemapModel (GameModel)

5.1.6 Klasse Challenge

Beschreibung:

Eigenschaften:

public Knot Start

public Knot Target

private SortedList¡Integer, String¿ highscore

public String Name

private IChallengeIO file

public IEnumerator¡KeyValuePair¡String, Integer¿¿ Highscore

public ChallengeMetaData Info

Methoden:

public ChallengeInfo Challenge (ChallengeInfo infoChallenge)

public Challenge Challenge (ChallengeIChallangeIO file)

public Boolean CreateChallenge (BooleanKnot startKnot targetString nameIChallengeIO file)

public String AddToHighscore (String nameInteger time)

5.1.7 Klasse ChallengeFileIO

Beschreibung:

Eigenschaften:

public IEnumerable¡KeyValuePair¡String, Integer¿¿ Highscore

public String Name

public Knot StartKnot

public Knot TargetKnot

private KnotStringIO startParser

private KnotStringIO targetParser

public ChallengeMetaData Meta

Methoden:

public String ChallengeFileIO (String path)

public Challenge Save (Challenge challenge)

5.1.8 Klasse ChallengeMetaData

Beschreibung:

Eigenschaften:

public String Name

public KnotMetaData Start

public KnotMetaData Target

public IChallengeIO File

5.1.9 Klasse ChallengeMode

Beschreibung:

Eigenschaften:

public void PlayerKnot

public void ChallengeKnot

public World PlayerKnot

private World PlayerWorld

private ModelRenderer ChallengeKnotRenderer

private ModelRenderer PlayerKnotRenderer

private PipeMovement PlayerKnotMovement

Methoden:

public void Update ()

5.1.10 Klasse Circle

Beschreibung:

Eigenschaften:

public T content

public Circle next

public Circle previous

Methoden:

public T Circle (T content)

5.1.11 Klasse ColorPicker

Beschreibung:

Eigenschaften:

public Color Color

Methoden:

public void OnKeyEvent ()

public Rectangle Bounds (Rectangle)

public void OnLeftClick ()

public void OnRightClick ()

5.1.12 Klasse ColorPickItem

Beschreibung:

Eigenschaften:

public Color Color

5.1.13 Klasse ConfirmDialog

Beschreibung:

5.1.14 Klasse ControlSettingsScreen

Beschreibung:

Eigenschaften:

protected void SettingsMenuSection

Methoden:

public void Update ()

5.1.15 Klasse CreativeMode

Beschreibung:

Eigenschaften:

public void Knot

public World Knot

private ModelRenderer KnotRenderer

Methoden:

public void Update ()

5.1.16 Klasse CreditsScreen

Beschreibung:

Methoden:

public void Update ()

5.1.17 Klasse Dialog

Beschreibung:

Eigenschaften:

public String Name

public String Text

Methoden:

public void OnKeyEvent ()

public Rectangle Bounds (Rectangle)

public void OnLeftClick ()

public void OnRightClick ()

5.1.18 Klasse DrawableGameStateComponent

Beschreibung:

Eigenschaften:

public GameScreen State

public DisplayLayer Index

Methoden:

public IEnumerable SubComponents (GameTimeIEnumerable GameTime)

public void DrawableGameStateComponent (GameScreenDisplayLayer)

5.1.19 Klasse DropDownMenuItem

Beschreibung:

Methoden:

public void AddEntries (DropDownEntrie)

5.1.20 Klasse Edge

Beschreibung:

Eigenschaften:

public IEnumerable Edges

public String Name

public KnotMetaData Meta

Methoden:

public IEnumerator¡Edge¿ GetEnumerator (IEnumerator¡Edge¿)

public Knot Save (Knot knot)

public string PrinterIO (string path)

5.1.21 Klasse Edge

Beschreibung:

5.1.22 Klasse Edge

Beschreibung:

Eigenschaften:

public Boolean Selected

public Color EdgeColor

public Direction Dir

public List¡int¿ Rectangles

Methoden:

public Direction Edge (Direction dir)

public Vector3 Get3DDirection (Vector3)

5.1.23 Klasse FadeEffect

Beschreibung:

Eigenschaften:

private bool IsFinished

private RenderTarget2D PreviousRenderTarget

Methoden:

public void FadeEffect (GameScreenGameScreen)

protected void DrawRenderTarget (GameTime)

5.1.24 Klasse FileIO

Beschreibung:

Eigenschaften:

public String FileName

Methoden:

public String ConvertToFileName (StringString)

5.1.25 Klasse Game

Beschreibung:

5.1.26 Klasse Game

Beschreibung:

Methoden:

public void Update ()

5.1.27 Klasse GameModelInfo

Beschreibung:

Eigenschaften:

public string Modelname

public Angles3 Rotation

public Vector3 Scale

Methoden:

public void GameModelInfo (String)

5.1.28 Klasse GameObjectInfo

Beschreibung:

Eigenschaften:

public bool IsMovable

public bool IsSelectable

public bool IsVisible

public Vector3 Position

Methoden:

public bool Equals (GameObjectInfobool GameObjectInfo)

5.1.29 Klasse GameScreen

Beschreibung:

Eigenschaften:

public Knot3Game Game

public InputHandler Input

public RenderEffect PostProcessingEffect

Methoden:

public void Entered ()

public void BeforeSwitch ()

public void BeforeExit ()

public void Update ()

public void Entered (Game)

public void AddGameComponents ()

public void RemoveGameComponents ()

5.1.30 Klasse GameScreenManager

Beschreibung:

Methoden:

public void Peek ()

public void Switch (GameScreen)

public void Pop (GameScreen)

public void Push (GameScreen)

5.1.31 Klasse GameStateComponent

Beschreibung:

Eigenschaften:

public DisplayLayer Index

public GameScreen State

Methoden:

public IEnumerable SubComponents (GameTimeIEnumerable GameTime)

public void GameStateComponent (GameScreenDisplayLayer)

5.1.32 Klasse GraphicsSettingsScreen

Beschreibung:

Eigenschaften:

protected void SettingsMenuSection

Methoden:

public void Update ()

5.1.33 Klasse IGameObject

Beschreibung:

Eigenschaften:

public GameObjectInfo Info

public World World

public float Alpha

public Color BaseColor

public Color HightlightColor

public float HighlightIntensity

public GameModelInfo Info

public XNA.Model Model

public World World

public Matrix WorldMatrix

Methoden:

public Vector3 Center (Vector3)

public void Update (GameTime)

public void Draw (GameTime)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public Vector3 Center (Vector3)

public void Update (GameTime)

public void Draw (GameTime)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public void GameModel (GameScreenGameModelInfo)

5.1.34 Klasse IGameObject

Beschreibung:

5.1.35 Klasse InputItem

Beschreibung:

Eigenschaften:

public String InputText

5.1.36 Klasse Knot

Beschreibung:

Eigenschaften:

public String Name

private Circle edges

public KnotMetaData Info

private IKnotIO file

Methoden:

public void Knot ()

public void Save ()

public IKnotIO Knot (IKnotIO file)

public KnotMetaData Knot (KnotMetaData info)

public Boolean IsValidMove (BooleanDirection dirInteger distance)

public Boolean Move (BooleanDirection dirInteger distance)

public Knot (Knot knotAKnot knotBBoolean)

public Boolean (BooleanKnot knotAKnot knotB)

public IEnumerator¡Edge¿ GetEnumerator (IEnumerator¡Edge¿)

public IKnotInfo Save (IKnotInfo file)

5.1.37 Klasse Knot3Game

Beschreibung:

Eigenschaften:

public bool IsFullScreen

public GameScreenManager Screens

public bool VSync

public GraphicsDeviceManager Graphics

Methoden:

public void Game ()

public void Initialize ()

public void LoadContent ()

public void UnloadContent ()

public void Draw (GameTime)

public void Game (GameTime)

5.1.38 Klasse KnotFileIO

Beschreibung:

Eigenschaften:

public IEnumerable¡Edge¿ Edges

public String Name

private KnotStringIO parser

public KnotMetaData Meta

Methoden:

public String KnotFileIO (String path)

public Knot Save (Knot knot)

5.1.39 Klasse KnotMetaData

Beschreibung:

Eigenschaften:

public String Name

public IKnotIO File

public Integer CountEdges

Methoden:

protected KnotMetaData KnotInfo (KnotMetaDataString nameInteger countEdgesIKnotIO(file)

5.1.40 Klasse KnotStringIO

Beschreibung:

Eigenschaften:

public String Name

public IErnumerable¡Edge¿ Edges

public String Content

public KnotMetaData Meta

Methoden:

public Knot Save (Knot knot)

public String KnotStringIO (String content)

5.1.41 Klasse LoadScreen

Beschreibung:

Methoden:

public void Update ()

5.1.42 Klasse Menu

Beschreibung:

Eigenschaften:

public String Name

Methoden:

public void Add (MenuItemMenuItem)

public void Delete (MenuItemMenuItem)

public void GetChild (intint)

public int Size (int)

public IEnumerator¡MenuItem¿ GetEnumerator (IEnumerator¡MenuItem¿)

5.1.43 Klasse MenuButton

Beschreibung:

Eigenschaften:

public String Name

5.1.44 Klasse MenuItem

Beschreibung:

Methoden:

public void OnLeftClick ()

public void OnRightClick ()

public void OnKeyEvent ()

public Rectangle Bounds (Rectangle)

5.1.45 Klasse MenuItem

Beschreibung:

Eigenschaften:

public String Text

Methoden:

public IEnumerator¡MenuItem¿ GetEnumerator (IEnumerator¡MenuItem¿)

5.1.46 Klasse MenuItem

Beschreibung:

5.1.47 Klasse MenuScreen

Beschreibung:

Methoden:

public void Update ()

5.1.48 Klasse ModelRenderer

Beschreibung:

Eigenschaften:

public GameObjectInfo Info

public World World

private List¡ArrowModel¿ arrows

private List¡NodeModel¿ nodes

private List¡PipeModel¿ pipes

public Knot Knot

Methoden:

public Vector3 Center (Vector3)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public void OnEdgesChanged ()

public void ModelRenderer (GameStateGameObjectInfoGameState)

public void Update (GameTime)

public void Draw (GameTime)

public IEnumerator GetEnumerator (IEnumerator)

5.1.49 Klasse MousePointer

Beschreibung:

Methoden:

public void MousePointer (GameState)

public void Draw (GameTime)

5.1.50 Klasse NodeModel

Beschreibung:

Eigenschaften:

public NodeModelInfo Info

Methoden:

public void NodeModel (GameScreenNodeModelInfo)

public void Draw (GameTime)

public void Update (GameTime)

5.1.51 Klasse NodeModelInfo

Beschreibung:

Eigenschaften:

public void EdgeFrom

public void EdgeTo

public void Knot

public Vector3 EdgeFrom

Methoden:

public void NodeModelInfo (EdgeListEdgeEdge)

5.1.52 Klasse PipeModel

Beschreibung:

Eigenschaften:

public PipeModelInfo Info

Methoden:

public void Draw (GameTime)

public void Update (GameTime)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public void PipeModel (GameScreenPipeModelInfo)

5.1.53 Klasse PipeModelInfo

Beschreibung:

Eigenschaften:

public Edge Edge

public Knot Knot

public Vector3 PositionFrom

public Vector3 PositionTo

Methoden:

public void PipeModelInfo (EdgeListEdge)

5.1.54 Klasse PipeMovement

Beschreibung:

Eigenschaften:

public GameObjectInfo Info

public Knot Knot

public World World

Methoden:

public Vector3 Center (Vector3)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public void Update (GameTime)

public void PipeMovement (GameStateWorldGameObjectInfo)

public IEnumerator GetEnumerator (IEnumerator)

public void Draw (GameTime)

5.1.55 Klasse ProfileSettingsScreen

Beschreibung:

Eigenschaften:

protected void SettingsMenuSection

public string Name

Methoden:

public void Update ()

5.1.56 Klasse RenderEffect

Beschreibung:

Eigenschaften:

public RenderTarget2D RenderTarget

protected GameScreen screen

protected SpriteBatch spriteBatch

Methoden:

public void Begin (GameTime)

public void End (GameTime)

public void DrawModel (GameModelGameTime)

public void RemapModel (GameModel)

protected void DrawRenderTarget (GameTime)

5.1.57 Klasse RenderEffectStack

Beschreibung:

Eigenschaften:

public IRenderEffect CurrentEffect

private IRenderEffect DefaultEffect

Methoden:

public void ()

public void (IRenderEffect)

public void RenderEffectStack ()

5.1.58 Klasse SaveLoadDialog

Beschreibung:

5.1.59 Klasse SettingsScreen

Beschreibung:

Eigenschaften:

protected void SettingsMenuSection

Methoden:

public void Update ()

5.1.60 Klasse ShadowGameModel

Beschreibung:

Eigenschaften:

public Color ShadowColor

public float ShadowAlpha

Methoden:

public void ShadowGameModel (GameStateGameModel)

public void Draw (GameTime)

5.1.61 Klasse ShadowGameObject

Beschreibung:

Eigenschaften:

public GameObjectInfo Info

public World World

public Vector3 ShadowPosition

public Vector3 OriginalPosition

Methoden:

public Vector3 Center (Vector3)

public void Update (GameTime)

public void Draw (GameTime)

public GameObjectDistance Intersects (RayGameObjectDistance Ray)

public void ShadowGameObject (GameStateIGameObject)

5.1.62 Klasse SliderItem

Beschreibung:

Eigenschaften:

public void MaxValue

public int Value

public int MinValue

5.1.63 Klasse SoundSlider

Beschreibung:

5.1.64 Klasse StandardEffect

Beschreibung:

Methoden:

protected void DrawRenderTarget (GameTime)

public void StandardEffect (GameScreen)

5.1.65 Klasse SubMenu

Beschreibung:

5.1.66 Klasse T

Beschreibung:

5.1.67 Klasse TextInputDialog

Beschreibung:

Eigenschaften:

public String InputText

5.1.68 Klasse TextItem

Beschreibung:

Eigenschaften:

public String Text

5.1.69 Klasse VerticalMenu

Beschreibung:

5.1.70 Klasse Widget

Beschreibung:

Eigenschaften:

public Vector2 RelativeSize

public Vector2 RelativePosition

public bool IsVisible

public Color BackgroundColor

public Color ForegroundColor

Methoden:

public Rectangle BoundingBox (Rectangle)

public void Widget ()

5.1.71 Klasse World

Beschreibung:

Eigenschaften:

public void Camera

public List¡IGameObject¿ Camera

public IGameObject SelectedObject

public IRenderEffect CurrentEffect

Methoden:

public void Update (GameTime)

public void Draw (GameTime)

public void World (GameScreen)

public IEnumerator¡IGameObject¿ GetEnumerator (IEnumerator¡IGameObject¿)

5.1.72 Klasse XNA.DrawableGameComponent

Beschreibung:

Methoden:

public void Draw (GameTime)

public void Update (GameTime)

5.1.73 Klasse XNA.Game

Beschreibung:

Methoden:

public void Game ()

public void Initialize ()

public void LoadContent ()

public void UnloadContent ()

public void Game (GameTime)

public void Draw (GameTime)

5.1.74 Klasse XNA.GameComponent

Beschreibung:

Methoden:

public void Update (GameTime)

Kapitel 6

Anmerkungen

Kapitel 7

Gloassar