Piraccini Yanick, Stadler Raphael

TBZ  M254

Tic tac toe Ai

Contents

[Zielsetzung 2](#_Toc125907228)

[Technologie 2](#_Toc125907229)

[Planung 2](#_Toc125907230)

[Use Case 3](#_Toc125907231)

# Zielsetzung

Die Entscheidung, eine Tic Tac Toe AI im Camunda Modeler zu implementieren, ist auf das Ziel zurückzuführen, dass wir den Prozess einer AI für den User bildlich darstellen wollen, um so das Verständnis der Entscheidungen einer AI dem User näherbringen.

Dies wird nicht einfach, da eine Tic Tac Toe AI schon an sich sehr komplex ist und von viele Algorithmen Gebrauch macht, jedoch wollen wir uns dieser Aufgabe annehmen und so gut wie möglich eine Tic Tac Toe AI in Camunda Modeler nachzubilden.

# Technologie

Da wir zuvor noch nie etwas mit dem Camunda Modeler gemacht hatten mussten wir uns zuerst etwas über den Aufbau informieren, leider gab es dazu einen nicht allzu ausführliche Doku bei der wir nur das Nötigste erfahren konnten. Ein grosses Hindernis war au das wir die Modeler Sprache 7 gebrauchten und nicht die neuste Version 8. Für das UI haben wir uns für WPF entschieden da wir uns im Bereich C# schon gut auskennen und so uns kein neues Wissen aneignen mussten. Um die Informationen vom Frontend zum Backend zu schicken, werden wir von ASP.NET Gebrauch machen da wir schon oft mit dem server-side web-application framework gearbeitet haben.

Aufbau:

* Frontend
  + WPF
  + ASP.NET
* Backend
  + Camunda Modeler
    - Gateway Interface

# Planung

|  |  |  |
| --- | --- | --- |
| Datum | Tätigkeit | Bemerkungen |
| 05.12.2022 | Vorstellung LB2 Use Case Definition | Wir entschieden uns für ein Thema und erstellten dazu 3 Use Cases die wir Umsetzen wollten. |
| 12.12.2022 | Prozess Definition und Spezifikation, Identifikation der Task- und Subtask Komponenten.  User Interface Design und API- Definitionen | Wir haben uns über die Komplexität einer Tic Tac Toe AI informiert und uns Gedanken dazu gemacht, wie wir dies in einem Bpmn umsetzten wollen. |
| 19.12.2022 | UI | Wir haben mit WPF einen simples Frontend gemacht auf dem man Tic Tac Toe spielen kann. |
| 09.01.2023 | Camunda Modeler backend | Wir haben im Camunda Modeler angefangen das Backend zu erstellen, jedoch hatten wir grosse Schwierigkeiten da wir zuvor noch nie einen REST API in einem Modeler gemacht haben |
| 16.01.2023 | Kommunikation  Frontend -> backend | Wir haben mit ASP.NET die Kommunikation vom Frontend zum Backend gelöst jedoch mussten wir das gleiche auch für das Backend machen. Dort konnten wir dies mit ServiceTasks machen dies hat uns jedoch etwas Zeit gekostet dies herauszufinden. |
| 23.01.2023 | Camunda Modeler backend | Fertigstellen des backends. |
| 30.01.2023 | Abgabe der Arbeit, Live Demo im Klassenverband. | Steht noch bevor |

# Use Case

|  |  |  |
| --- | --- | --- |
| **Name** | **Spiel starten** | |
| Akteur | Spieler | |
| Trigger | Starten der Applikation | |
| Kurzbeschreibung | Der Spieler startet die Desktop Applikation und ihm wird das Spielfeld angezeigt auf der, der Computer schon ein X platziert hat. | |
| Vorbedingungen | Camunda Platform Run muss laufen unter <http://localhost:8080>, Es muss eine Prozess Instance auf dem Camunda Platform Run vorhanden sein | |
| Essenzielle Schritte | ***Intention der Systemumgebung*** | ***Reaktion des Systems*** |
| Anzeigen des Spielfeldes | Laden der XAML-Datei |
| Web Api aufstarten | Die ASP.NET Web API wird aufgestartet und schickt einen request um den Camunda Prozess zu starten. |
| Starten des Backend Prozesses | Es wird ein POST request an das Backend geschickt in dem der Prozess gestartet wird |
| Platzieren des ersten X durch Computer | Nach dem Starten wird ein Int mit dem Standort des ersten X’s an das Frontend zurückgesendet |
|  |  |
|  |  |
|  |  |
| Ausnahmefälle | keine |  |
| Nachbedingung | keine |  |
| Zeitverhalten | keine |  |
| Verfügbarkeit | 7/24 |  |
| Folge Use Case | Der Spieler kann sein erstes O plazieren |  |

|  |  |  |
| --- | --- | --- |
| **Name** | Spieler setzt O | |
| Akteur | Spieler | |
| Trigger | Drücken eines Beliebigen Felds | |
| Kurzbeschreibung | Der Spieler hat nun die Möglichkeit ein O auf dem Spielfeld zu setzten | |
| Vorbedingungen | Applikation gestartet, Camunda Platform Run muss laufen unter <http://localhost:8080> | |
| Essenzielle Schritte | ***Intention der Systemumgebung*** | ***Reaktion des Systems*** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Ausnahmefälle | keine |  |
| Nachbedingung | keine |  |
| Zeitverhalten | sofort |  |
| Verfügbarkeit | 7/24 |  |
| Folge Use Case | Backend Check für möglicher Win |  |

|  |  |  |
| --- | --- | --- |
| **Name** | **Überprüfen für Computer Win chance** | |
| Akteur | kein | |
| Trigger | 2 X wurden platziert | |
| Kurzbeschreibung | Wenn 2 X platziert wurde, wird überprüft ob es möglich ist für den Computer zu gewinnen | |
| Vorbedingungen | Applikation gestartet, Camunda Platform Run muss laufen unter <http://localhost:8080> | |
| Essenzielle Schritte | ***Intention der Systemumgebung*** | ***Reaktion des Systems*** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Ausnahmefälle | keine |  |
| Nachbedingung | keine |  |
| Zeitverhalten | sofort |  |
| Verfügbarkeit | 7/24 |  |
| Folge Use Case | Backend Check für möglicher Win |  |

|  |  |  |
| --- | --- | --- |
| **Name** | **Überprüfen auf Spieler WIn** | |
| Akteur | Spieler | |
| Trigger | Drücken eines Beliebigen Felds | |
| Kurzbeschreibung | Der Spieler hat nun die Möglichkeit ein O auf dem Spielfeld zu setzten | |
| Vorbedingungen | Applikation gestartet, Camunda Platform Run muss laufen unter <http://localhost:8080> | |
| Essenzielle Schritte | ***Intention der Systemumgebung*** | ***Reaktion des Systems*** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Ausnahmefälle | keine |  |
| Nachbedingung | keine |  |
| Zeitverhalten | sofort |  |
| Verfügbarkeit | 7/24 |  |
| Folge Use Case | Backend Check für möglicher Win |  |