

# Covid19-Screening

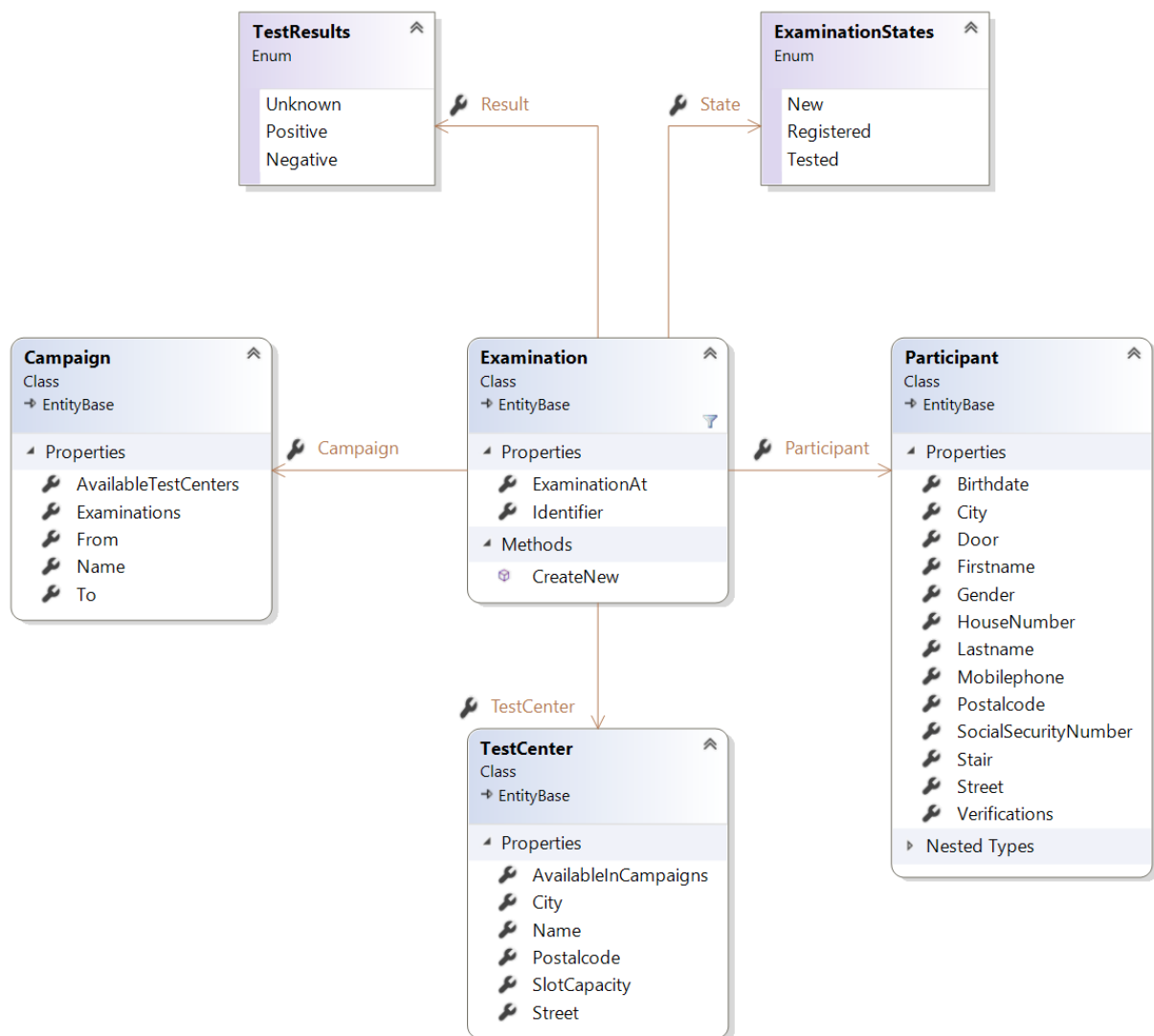
Um die österreichische Regierung während der Covid19-Pandemie zu unterstützen soll eine Software zur Verwaltung der großflächig angelegten Covid19-Testungen umgesetzt werden.

## Anforderungen

### Datenmodell

- Es sollen Kampagnen (**Campaign**) für Testungen (**Examination**) erstellt werden können. Eine solche Kampagne hat ein definiertes Beginn- und ein definiertes Ende-Datum. Beispiele solcher Kampagnen sind:
  - o Antigentest Bildungsbereich OÖ (01.10.2020 – 14.10.2020)
  - o Antigentest Bildungsbereich Wien (03.10.2020 – 24.10.2020)
  - o Antigentest Gesamtbevölkerung Oberösterreich (15.10.2020 – 05.11.2020)
  - o Antigentest Gesamtbevölkerung Niederösterreich (15.10.2020 – 30.10.2020)
  - o Antigentest Gesundheitsbereich Steiermark (10.10.2020 – 17.10.2020)
  - o usw.
- Für die Durchführung der Tests werden **TestCenter** benötigt, welche ebenfalls über die Software verwaltet werden sollen.
- Diese **TestCenter** müssen für die einzelnen Kampagnen freigeschaltet werden.
- Um den Strom der zu testenden Personen zu lenken wird für das TestCenter die maximale Anzahl der gleichzeitigen Testungen pro 15 Minuten festgelegt. Es können sich nie mehr Personen für einen solchen Slot anmelden als festgelegt.
- Eine Person (**Participant**), welche sich für eine oder mehrere Testungen registriert, wird nur einmal im System verwaltet. Für diese werden die wichtigsten Stammdaten einmalig gespeichert.
- Eine Testung (**Examination**) kann von einer Person (**Participant**) über eine Web-Seite angemeldet werden sowie ein Termin für die Testung reserviert werden. Eine Testung kann folgende Zustände haben: Neu, Registriert, Getestet (**ExaminationStates**). Das Ergebnis der Testung muss ebenfalls persistiert werden: Positiv, Negativ und Unbekannt (->wenn Testung noch nicht stattgefunden hat bzw. die Person zur Testung nicht erschienen ist) (**TestResults**).
- Damit eine Person eindeutig identifiziert werden kann muss ihre Sozialversicherungsnummer bekannt sein.
- Das Testergebnis soll per SMS zugestellt werden. Somit ist die Speicherung/Verarbeitung der Handynummer der Person erforderlich.
- Jede Person bekommt bei der Reservierung eines Termins (**Examination**) eine zufällig generierte Identifikationsnummer zugeordnet. Verwenden Sie dazu folgendes NuGet-Paket: <https://github.com/sarthakmahapatra/StringRandomizer>.  
Mit dieser Identifikationsnummer kann zum einen die Reservierung zur Testung über die Web-Seite geändert werden (Zeitpunkt, TestCenter, etc.). Zum anderen kann sich die Person mit dieser Identifikationsnummer beim **TestCenter** vor Ort für die Testung „ausweisen“.

Ein beispielhaftes Klassendiagramm könnte folgendermaßen aussehen:



## Sicherheit

- Sicherheit ist ein wichtiges Thema. Deshalb muss sich eine Person auf der Web-Seite mit ihrer Handynummer verifizieren. Dazu wird ein zufälliger Token (6 Zufallszahlen, z.B. „645239“) generiert und per SMS an die von der Person angegebenen Handynummer übermittelt werden. Wenn die Person den korrekten Token auf der Web-Seite angibt, so kann von einer erfolgreichen Verifizierung der Person ausgegangen werden.
- Dieser Token ist nur eine gewisse Zeit gültig (15 Minuten)!
- Die Token-Identifikation muss von Seite zu Seite (Anmeldung im System -> Terminreservierung -> Wartung der persönlichen Daten) in der URL weitergereicht werden und bei jedem (!) Seitenaufruf validiert werden.

## Umsetzung (Teil 1)

Setzen Sie eine Web Seite mithilfe von Razor Pages um, welche den folgenden Anforderungen genügt:

- Bereitstellung einer Web-Seite auf welcher sich Personen (**Participant**) registrieren können.
- Nach der Registrierung kann sich die Person zu einer Testung (**Examination**) anmelden/einen Termin für einen freien Slot reservieren.
- Reservierungsbestätigung per SMS-Versand:
  - Identifikation-Nummer
  - Ausgewählter Zeitpunkt der Testung
  - Daten zum TestCenter
- Eine Person kann eine angemeldete Testung auch adaptieren. Es kann ein anderes Test-Center ausgewählt werden bzw. ein anderer Termin gewählt werden.
- Eine Person kann auf der Web-Seite ihre Stammdaten (Adresse, etc.) selbst verwalten.
- Stellen Sie in der Anwendung Testdaten bereit (Stichwort „ImportConsole“).
- Implementieren Sie die notwendigen Validierungen auf den Formularen

### Tipps:

- Zeichenkette
  - Verpflichtend (ja/nein)
  - Mindestlänge, Maximallänge
- Telefonnummer
  - Verpflichtend (ja/nein)
  - Format
- Sozialversicherungsnummer
  - Verpflichtend (ja/nein)
  - Validierung der Sozialversicherungsnummer lt. folgender Regel:

Länge	Struktur
10	lfd. Nr. <b>PZ</b> Geb.datum $x_1 x_2 x_3$ p T T M M J J
$x_1 \neq 0$ Der Monat kann > 12 werden (bei Überlauf, fehlendes Geb.datum etc.).[1]	

- Die Ziffernfolge wird von links nach rechts mit 3, 7, 9, 5, 8, 4, 2, 1, 6 gewichtet.
- Die Produkte werden summiert.
- Von der Summe wird der volle Rest zur nächst niedrigeren durch 11 teilbaren Zahl (modulo 11) bestimmt.
- Ist der Rest 10, wird die Nummer nicht vergeben.

Beispiel 328p171076

Nummer	Schritt 1: Gewichtung	Schritt 2: Produkt- sumation
3	3	9
2	7	14
8	9	72
p		
1	5	5
7	8	56
1	4	4
0	2	0
7	1	7
6	6	36
Summe		203
Schritt 3: Summe mod 11		$203 \div 11 = 18 \text{ Rest } 5$
Endergebnis Prüfziffer		5
3285171076		

Quelle: [Prüfziffern: Sozialversicherungsnummer \[AT\] \(pruefziffernberechnung.de\)](#)

- Ganzzahlige Datentypen
    - Verpflichtend (ja/nein)
    - Wertebereich (von-bis)
  - Andere Datentypen
    - Verpflichtend (ja/nein)
    - Format
- Da es sich um eine öffentlichkeitswirksame Web-Seite handelt muss das Erscheinungsbild einen modernen Character aufweisen.

Tipp:

Verwenden Sie Bootstrap um die Web-Seite bzw. Formulare optisch ansprechend zu gestalten.