

# Cuprins

<b>1</b>	<b>INTRODUCERE</b>	<b>5</b>
1.1	Scopul principal . . . . .	5
1.2	Motivație . . . . .	5
<b>2</b>	<b>TEHNOLOGII ȘI FRAMEWORK-URI FOLOSITE</b>	<b>6</b>
2.1	C# . . . . .	6
2.2	.NET . . . . .	7
2.3	Azure Function . . . . .	7
2.3.1	Azure Durable Functions . . . . .	8
2.4	Azure Table Storage . . . . .	9
2.4.1	Azurite . . . . .	10
2.5	LINQ . . . . .	10
2.6	HyperText Markup Language (HTML) . . . . .	11
2.7	Cascading Style Sheets (CSS) . . . . .	12
2.8	Bootstrap . . . . .	12
2.9	Blazor . . . . .	13
2.10	MudBlazor . . . . .	15
2.11	SignalR . . . . .	15
2.11.1	SignalR Server Emulator . . . . .	16
2.12	Refit API . . . . .	17
2.13	Leaflet și Routing Machine . . . . .	17
2.14	Stripe . . . . .	18
<b>3</b>	<b>ELEMENTE SOFTWARE FOLOSITE</b>	<b>20</b>
3.1	JetBrains Rider IDE . . . . .	20
3.2	Microsoft Azure Storage Explorer . . . . .	20
3.3	Docker . . . . .	21
<b>4</b>	<b>FLUXUL APLICAȚIEI</b>	<b>22</b>
4.1	Experiența utilizatorului . . . . .	24
4.2	Experiența șoferului . . . . .	29
4.3	Experiența admin-ului . . . . .	30
<b>5</b>	<b>ARHITECTURA APLICAȚIEI</b>	<b>32</b>
5.1	Comunicarea între componente . . . . .	32
5.2	Stocare . . . . .	37

5.3	Server-side . . . . .	39
5.4	Client-side . . . . .	50
5.5	Instalarea aplicației . . . . .	57
<b>6</b>	<b>CONCLUZII</b>	<b>59</b>

## Listă de figuri

4.1	Diagrama <i>Use case</i> a proiectului. . . . .	23
4.2	Bara de navigare pe desktop și mobile. . . . .	24
4.3	Mark-ul <i>human</i> ce indică locația actuală a utilizatorului. . . . .	24
4.4	Mark-ul <i>pin</i> pentru a sugera destinația. . . . .	24
4.5	Bara de căutare pentru adrese cu sugestii. . . . .	25
4.6	Mark-ul <i>driver</i> ce semnifică un utilizator de tip șofer disponibil. . . . .	25
4.7	Pașii pe care utilizatorul trebuie să îi completeze. . . . .	26
4.8	Mesajul de confirmare că totul este în regulă. . . . .	26
4.9	Meniul de informare al status-ului. . . . .	27
4.10	Detaliile despre șofer după ce acceptă cursa. . . . .	27
4.11	Mark-ul <i>currentCar</i> ce semnifică că utilizatorul călătorește cu mașina. . . . .	27
4.12	Dialog-ul ce permite oferirea unui recenzii șoferului. . . . .	28
4.13	Schimbarea numărului de telefon apăsând pe imaginea de profil. . . . .	28
4.14	Istoricul curselor. . . . .	29
4.15	Bara de navigare împreună cu dialogul-ul ce afișează ofertele curente. . . . .	29
4.16	Exemplu de rută și indicații de navigare. . . . .	30
4.17	Pagina administratorului. . . . .	30
5.1	Comunicarea dintre componentele proiectului. . . . .	32
5.2	Metoda ce obține rezultatul dorit în urma execuției endpoint-ului. . . . .	33
5.3	Interfața Refit. . . . .	34
5.4	Înregistrarea DelegatingHandler-ului din Client ( <i>THttpDelegatingHandler</i> este <i>HttpAuthenticationHandler</i> ) și adăugarea header-urilor de autentificare și autorizare. . . . .	34
5.5	Conectarea la server-ul SignalR și crearea subscripției pentru notificarea noii locații a unui șofer pe hartă. . . . .	35
5.6	Trimiterea notificării către utilizator că s-a ajuns la unul din checkpoint-uri. . . . .	35
5.7	Atributul ce atribuie entități un nume de tabel pentru storage . . . . .	36
5.8	Cum se folosește atributul <i>TableName</i> pe entitatea <i>userEntity</i> . . . . .	36
5.9	Diagrama ER pentru baza de date. . . . .	37
5.10	Exemplu de metodă GET pentru <i>HttpTrigger</i> function. . . . .	40
5.11	Deserializarea unui request într-un tip concret (ex: <i>UserIdentifier</i> ). . . . .	40
5.12	Maparea unui <i>ServiceResponse</i> la <i>IActionResult</i> . . . . .	41
5.13	Verificare <i>AccessToken</i> folosind Google API. . . . .	42
5.14	Configurația Google Credentials. . . . .	42

5.15	TimeTrigger function ce interoghează Durable tables și notifică utilizatorii despre curse lor ce sunt în desfășurare. . . . .	43
5.16	Înscrierea și ștergerea unui utilizator din grup. . . . .	44
5.17	<i>Negotiate</i> , conectarea și deconectarea unui utilizator de la conexiune SignalR. .	45
5.18	Salvarea în <i>CustomStatus</i> a request-ului inițial. . . . .	46
5.19	Trimiterea confirmării către utilizator și așteptarea răspunsului. . . . .	46
5.20	Calcularea prețului cursei. . . . .	46
5.21	Crearea <i>PaymentIntent</i> -ului prin Stripe. . . . .	47
5.22	Așteptarea răspunsului de la șofer și cronometrarea timpului de acceptare. . . .	47
5.23	Anularea cursei dacă plata nu s-a procesat, a fost intenționat anulată sau nu s-a găsit un șofer. . . . .	48
5.24	Declararea activității ce pune orchestration-ul să aștepte $n$ secunde. . . . .	49
5.25	Filtrarea log-urilor din Azure. . . . .	50
5.26	Conținutul folderului <i>wwwroot</i> . . . . .	50
5.27	Inițializarea hărții după ce componenta s-a randat. . . . .	51
5.28	Dicționarul ce face legătura dintre tipul de marker și imaginea de îl reprezintă. .	52
5.29	Atașarea input-ului pentru card bancar de elementul cu ID-ul <i>payment-element</i> . .	52
5.30	Configurarea Stripe în dashboard-ul platformei și istoricul tranzacțiilor. . . . .	53
5.31	Execuția din 3 în 3 secunde a Background Service-ului. . . . .	54
5.32	<i>TaskCompletionSource</i> așteptând ca scriptul de JavaScript să trigger-uiască event-ul <i>CoordinatesChanged</i> . . . . .	54
5.33	Pagina de autentificare ce redirecționează utilizatorul către Google Login. . . . .	55
5.34	Înregistrarea autentificării și folosirea autorizării în aplicație. . . . .	56
5.35	Fișierul JSON ce conține coordonatele mock-uite pentru utilizatori. . . . .	56