

```

//Prijava korisnika
@startuml
hide footbox
title Login
actor Guide as Guide
participant "SignInViewModel" as LoginVM
participant "UserService" as UserSvc
participant "UserRepository" as UserRepo
    Guide -> LoginVM : SignIn()
    activate LoginVM
    LoginVM -> UserSvc : Authenticate(username, password)
    activate UserSvc
    UserSvc -> UserRepo : Authenticate(username, password)
    activate UserRepo
    UserRepo --> UserSvc : user
    deactivate UserRepo
    alt Invalid credentials (user == null)
        UserSvc --> LoginVM : user (null)
        LoginVM --> Guide : error
    else
        UserSvc --> LoginVM : user
        deactivate UserSvc
        LoginVM --> Guide : user
        deactivate LoginVM
    end
end
@enduml

//Racunanje najpopularnijeg jezika/lokacije i vracanje id-ja
@startuml
hide footbox
title TopRequested
participant "RecommendationsViewModel" as VM
participant "TourRequestService" as StatsSvc
participant "TourRequestRepository" as ReqRepo
VM -> StatsSvc : CalculateTopRequested(criteria)
activate StatsSvc
StatsSvc -> ReqRepo : CalculateTopRequested(criteria)
activate ReqRepo
ReqRepo --> StatsSvc : topIds : List<int>
deactivate ReqRepo
alt No data (topIds[0] == 0)
    StatsSvc --> VM : [0]
else Equally popular (topIds.size > 1)
    StatsSvc --> VM : [id1,id2,...]
else Most popular (topIds[0]!=0)
    StatsSvc --> VM : [id]
end
deactivate StatsSvc
@enduml
//Kreiranje ture

```

```

@startuml
hide footbox
title Create tour (with <<create>> + loops + invalid input guard)
actor Guide as G
participant "RecommendationsViewModel" as RVM
participant "TourService" as TS
participant "TourInstanceService" as TIS
participant "KeyPointService" as KS
participant "ImageService" as IS
participant "TourStartTimeService" as TSTS
participant "TimeSlotService" as TSS
participant "TourRepository" as TR
participant "TourInstanceRepository" as TIR
participant "KeyPointRepository" as KR
participant "ImageRepository" as IR
participant "TourStartTimeRepository" as TSTR
participant "TimeSlotRepository" as TSR
G -> RVM : Create()
activate RVM
' Guard provera
RVM -> RVM : canCreate = CanCreate()
alt Invalid input (canCreate == false)
    ' Prikaz grešaka i blokiranje dugmeta
    RVM -> RVM : ShowValidationErrors()\nDisable Create button
    RVM --> G : error
    deactivate RVM
else Valid input (canCreate == true)
    ' --- CreateTour() ---
    ' eksplicitno kreiranje Tour objekta
    create "Tour" as TourObj
    RVM -> TourObj : <<create>>
    RVM -> TS : Save(tour)
    activate TS
    TS -> TR : Save(tour)
    activate TR
    TR --> TS : tour
    deactivate TR
    TS --> RVM : tour
    deactivate TS
    ' --- CreateDateTimeAndInstances(tourId) ---
    RVM -> RVM : CreateDateTimeAndInstances(tourId)
    loop for each startTime in DateEntries
        ' kreiranje TourStartTime za svaku iteraciju
        create "TourStartTime" as TSTi
        RVM -> TSTi : <<create>>
        RVM -> TSTS : Save(startTime)
        activate TSTS
        TSTS -> TSTR : Save(startTime)
        activate TSTR
    end

```

```

TSTR --> TSTS : startTime
deactivate TSTR
TSTS --> RVM : startTime
deactivate TSTS
' kreiranje TourInstance povezane sa savedTime
create "TourInstance" as Tli
RVM -> Tli : <<create>>
RVM -> TIS : Save(tourInstance)
activate TIS
TIS -> TIR : Save(tourInstance)
activate TIR
TIR --> TIS : tourInstance
deactivate TIR
TIS --> RVM : tourInstance
deactivate TIS
' (opciono) kreiranje TimeSlot za termin
create "TimeSlot" as SlotI
RVM -> SlotI : <<create>>
RVM -> TSS : Save(timeSlot)
activate TSS
TSS -> TSR : Save(timeSlot)
activate TSR
TSR --> TSS : timeSlot
deactivate TSR
TSS --> RVM : timeSlot
deactivate TSS
end
' --- CreateKeyPoints(tourId) ---
RVM -> RVM : CreateKeyPoints(tourId)
loop for each keyPoint in KeyPoints
' kreiranje KeyPoint entiteta i upis
create "KeyPoint" as KP
RVM -> KP : <<create>>
RVM -> KS : Save(keyPoint)
activate KS
KS -> KR : Save(keyPoint)
activate KR
KR --> KS : keyPoint
deactivate KR
KS --> RVM : keyPoint
deactivate KS
end
' --- CreateImage(tourId) ---
RVM -> RVM : CreateImage(tourId)
loop for each image in Images
' kreiranje Image entiteta i upis
create "Image" as IMG
RVM -> IMG : <<create>>
RVM -> IS : Save(image)

```

```

    activate IS
    IS -> IR : Save(image)
    activate IR
    IR --> IS : image
    deactivate IR
    IS --> RVM : image
    deactivate IS
end
' --- Success + reset ---
' informativna poruka i reset forme
RVM -> RVM : ShowInfo("Tour created successfully!")
RVM -> RVM : ResetForm()
RVM --> G : success
deactivate RVM
end
@enduml
//FINALNI DIJAGRAM
@startuml
hide footbox
scale 0.8
title Create Recommended Tour — Master Flow
actor Guide as Guide
participant "SignInViewModel" as LoginVM
participant "UserService" as UserSvc
participant "UserRepository" as UserRepo
participant "ShellViewModel" as ShellVM
participant "StatisticsViewModel" as StatsVM
participant "RecommendationsViewModel" as RVM
participant "TourRequestService" as StatsSvc
participant "TourRequestRepository" as ReqRepo
participant "TourService" as TS
participant "TourInstanceService" as TIS
participant "KeyPointService" as KS
participant "ImageService" as IS
participant "TourStartTimeService" as TSTS
participant "TimeSlotService" as TSS
participant "TourRepository" as TR
participant "TourInstanceRepository" as TIR
participant "KeyPointRepository" as KR
participant "ImageRepository" as IR
participant "TourStartTimeRepository" as TSTR
participant "TimeSlotRepository" as TSR
' --- LOGIN ---
activate Guide
ref over Guide, LoginVM, UserSvc, UserRepo
    Login
end ref
deactivate Guide
' --- NAVIGACIJA ---

```

```

Guide -> ShellVM : Open()
activate ShellVM
ShellVM --> Guide : ShellView loaded
deactivate ShellVM
ShellVM -> StatsVM : NavigateTo(StatisticsViewModel)
activate StatsVM
StatsVM --> ShellVM : StatisticsView loaded
deactivate StatsVM
StatsVM -> StatsVM : CreateRecommendedTourCommand.Execute()
StatsVM -> RVM : NavigateTo(RecommendedViewModel)
activate RVM
RVM --> StatsVM : RecommendedView loaded
Guide -> RVM : SetCriteria(Language/Location)
RVM --> Guide : SelectedItem(Language/Location)
' --- TopRequested ---
ref over RVM, StatsSvc, ReqRepo
  TopRequested (see TopRequested diagram)
end ref
' --- Grananje po kandidatima ---
alt No data (service returned ids[0]=0)
  RVM -> RVM : ShowInfo("There is no tour requests in last 12 months in the system")
else Multiple candidates
  RVM -> RVM : ShowCandidatePicker(ids)
  Guide -> RVM : Chooseld()
  RVM -> RVM : SetRecommended(selectedId)
else Single top (one candidate)
  RVM -> RVM : SetRecommended(topId)
end
' --- Unos forme ---
Guide -> RVM : Enter name / description / dates / key points / images
RVM --> Guide : Completed tour creation form
' --- Create tour ref blok ---
ref over RVM, TS, TIS, KS, IS, TSTS, TSS, TR, TIR, KR, IR, TSTR, TSR
  Create tour
end ref
' --- Notification kreiranje i snimanje ---
participant NotificationService
participant NotificationRepository
participant Notification as N
create N
RVM -> N : <<create>>
RVM -> NotificationService : Save(notification)
activate NotificationService
NotificationService -> NotificationRepository : Save(notification)
activate NotificationRepository
NotificationRepository --> NotificationService : savedNotification
deactivate NotificationRepository
NotificationService --> RVM : savedNotification
deactivate NotificationService

```

```

' --- TourNotification kreiranje i snimanje ---
participant TourNotificationService
participant TourNotificationRepository
participant TourNotification as T
create T
RVM -> T : <<create>>
RVM -> TourNotificationService : Save(tourNotification)
activate TourNotificationService
TourNotificationService -> TourNotificationRepository : Save(tourNotification)
activate TourNotificationRepository
TourNotificationRepository --> TourNotificationService : savedTourNotification
deactivate TourNotificationRepository
TourNotificationService --> RVM : savedTourNotification
deactivate TourNotificationService
deactivate RVM
@enduml
//Prikaz slozenih tura -> Ovde prepraviti konstruktor za dto
@startuml
hide footbox
title Show complex tours
participant "MyTourRequestsViewModel" as VM
participant "ComplexTourRequestService" as ComplexSvc
participant "ComplexTourRequestRepository" as ComplexRepo
participant "TourRequestService" as ReqSvc
participant "TourRequestRepository" as ReqRepo
VM -> ComplexSvc : GetActualComplexToursIds()
activate ComplexSvc
ComplexSvc -> ComplexRepo : GetActualComplexToursIds()
activate ComplexRepo
ComplexRepo --> ComplexSvc : ids
deactivate ComplexRepo
ComplexSvc --> VM : ids
deactivate ComplexSvc
alt No complex tours (ids.size == 0)
    VM -> VM : Message: No pedding complex tour requests.
else ids.size != 0
    loop for each id in ids
        VM -> ReqSvc : GetRequestsByComplexId(id)
        activate ReqSvc
        ReqSvc -> ReqRepo : GetRequestsByComplexId(id)
        activate ReqRepo
        ReqRepo --> ReqSvc : tourRequests
        deactivate ReqRepo
        ReqSvc --> VM : tourRequests
        deactivate ReqSvc
    end
end
@enduml
//Racunanje dostupnih slotova

```

```

@startuml
hide footbox
title Show time slots
participant "AvailableTimeViewModel" as VM
participant "TimeSlotService" as TSS
participant "TimeSlotRepository" as TSR
participant "TourRequestService" as TRS
participant "TourRequestRepository" as TRR
VM -> VM : ShowAvailableDates()
' -- dohvat slotova --
VM -> VM : GetTimeSlots()
VM -> TSS : GetAvailableTimeSlots(startTime, endTime)
activate TSS
TSS -> TSR : GetAvailableTimeSlots(startTime, endTime)
activate TSR
TSR --> TSS : timeSlots
deactivate TSR
TSS --> VM : timeSlots
deactivate TSS
' -- dohvat zahteva --
VM -> VM : GetRequests()
VM -> TRS : GetAcceptedRequestById(complexId)
activate TRS
TRS -> TRR : GetAcceptedRequestById(complexId)
activate TRR
TRR --> TRS : tourRequests
deactivate TRR
TRS --> VM : tourRequests
deactivate TRS
' -- alternative / rendering --
alt No time slots (timeSlots is empty)
  VM -> VM : ShowInfo("Nema dostupnih termina u izabranom periodu")
else Compare and filter
  VM -> VM : CompareTime(tourRequests, timeSlots)
  loop for each slot in timeSlots
    VM -> VM : Overlaps(slot, tourRequests)
  end
end
@enduml
FINALNI

@startuml
hide footbox
title Accept Complex Tour Request — Master (refs: Login, Show complex tours, Show time slots)
' --- Login participants ---
actor Guide as Guide
participant "SignInViewModel" as LoginVM
participant "UserService" as UserSvc
participant "UserRepository" as UserRepo
' --- ViewModels / navigation ---

```

```

participant "ShellViewModel" as SVM
participant "TourRequestViewModel" as TRVM
participant "AvailableTimeViewModel" as AVVM
' --- Show complex tours participants ---
participant "ComplexTourRequestService" as ComplexSvc
participant "ComplexTourRequestRepository" as ComplexRepo
participant "TourRequestService" as ReqSvc
participant "TourRequestRepository" as ReqRepo
' --- Show time slots participants ---
participant "TimeSlotService" as TSS
participant "TimeSlotRepository" as TSR
participant "TourRequestService" as TRS
participant "TourRequestRepository" as TRR
' --- LOGIN ---
activate Guide
ref over Guide, LoginVM, UserSvc, UserRepo
    Login (see Login diagram)
end ref
deactivate Guide
' --- NAVIGACIJA ---
Guide -> SVM : Open()
activate SVM
SVM --> Guide : ShellView loaded
deactivate SVM
SVM -> TRVM : Navigate to TourRequestViewModel
activate TRVM
TRVM -> TRVM : SelectTab("Complex tours")
TRVM --> SVM : Complex tours views
' TRVM ostaje aktivan dok ne završi izbor i proveru CanAccept
' (nema dodatne deaktivacije ovde)
' --- Show complex tours (referenca) ---
ref over TRVM, ComplexSvc, ComplexRepo, ReqSvc, ReqRepo
    Show complex tours (see diagram)
end ref
Guide -> TRVM : Choose complex request part (requestId)
' TRVM je već aktivan
TRVM -> ReqSvc : CanAccept(requestId, guidId)
activate ReqSvc
ReqSvc -> ReqRepo : CanAccept(requestId, guidId)
activate ReqRepo
ReqRepo --> ReqSvc : toBeAccepted
deactivate ReqRepo
ReqSvc --> TRVM : toBeAccepted
deactivate ReqSvc
alt Cannot accept (canAccept == false)
    TRVM -> TRVM : ShowInfo("This request cannot be accepted")
    TRVM --> Guide : error
    deactivate TRVM
else Can accept

```



```

    TRVM -> AVM : Navigate to AvailableTimeViewModel
    activate AVM
    deactivate TRVM
end
' --- Show time slots (referenca) ---
ref over AVM, TSS, TSR, TRS, TRR
    Show time slots (see diagram)
end ref
alt No available slots
    AVM -> AVM : ShowInfo("No available slots for selected range")
    AVM --> Guide : back
    deactivate AVM
else Slots available
    Guide -> AVM : SelectSlot(selectedSlot)
    ' AVM je već aktivan
    AVM -> AVM : OnAccept(selectedSlot)
    ' --- Update tour request status/details ---
    AVM -> TRS : Update(tourRequest)
    activate TRS
    TRS -> TRR : Update(tourRequest)
    activate TRR
    TRR --> TRS : updated
    deactivate TRR
    TRS --> AVM : updated
    deactivate TRS
    ' --- Create new TimeSlot explicitly (<<create>>) and save ---
    create "TimeSlot" as NewSlot
    AVM -> NewSlot : <<create>>
    AVM -> TSS : Save(slot)
    activate TSS
    TSS -> TSR : Save(slot)
    activate TSR
    TSR --> TSS : slot
    deactivate TSR
    TSS --> AVM : slot
    deactivate TSS
    ' --- Final feedback ---
    AVM -> AVM : ShowInfo("Request accepted")
    AVM --> Guide : success
    deactivate AVM
end
@enduml

```