

**XAMARIN.FORMS ОКВИР ЗА РАЗВОЈ МОБИЛНИХ АПЛИКАЦИЈА****XAMARIN.FORMS CROSS-PLATFORM FRAMEWORK**

Милош Стевановић, Факултет техничких наука, Нови Сад

**Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО**

**Кратак садржај** – Тема мастер рада спада у дисциплину програмирања мобилних апликација. У мастер раду је приказан увод у развој вишеплатформских апликација и употребу *Xamarin.Forms* алата за њихов развој. Приказан је развој мобилне апликације коришћењем *Xamarin.Forms* алата као и опис развијене апликације.

**Abstract** – *The subject of the master thesis belongs to the discipline of mobile application development. The thesis contains introduction to cross-platform development of mobile applications and usage of Xamarin.Forms tool. It is shown process of mobile application development using Xamarin.Forms tool.*

**1. УВОД**

Да би мобилна апликација била доступна што већем броју корисника потребно је да буде развијена за више платформи. То углавном није једноставан задатак зато што свака од популарних платформи (*Android*, *iOS* и *Windows*) подржава само апликације које су развијене за њу. *Cross platform* оквири за израду мобилних апликација омогућавају да једном развијена мобилна апликација може да се покрене на свим популарним платформама. Постоји велики број *cross platform* оквира, а један од најпопуларнијих је *Xamarin*.

**2. XAMARIN.FORMS**

*Xamarin.Forms* је вишеплатформско окружење за развој мобилних апликација за *Android* [1], *iOS* [2] и *Windows* [3] платформе користећи програмски језик *C#* [4]. Представља сет алата који омогућавају развој заједничког корисничког интерфејса који се дефинише на једном месту али функционише на све три платформе. Свака платформа дефинише своје матичне компоненте корисничког интерфејса због чега оне изгледају и понашају се исто као и матичне компоненте на конкретним платформама. Апликације развијене коришћењем *Xamarin.Forms* алата могу да имају све матичне карактеристике и да истовремено деле заједнички програмски код, логику апликације.

**НАПОМЕНА:**

Овај рад проистекао је из мастер рада чији ментор је био др Стеван Гостојић, ванр. проф.

**2.1 Како Xamarin.Forms функционише**

Све *Xamarin* библиотеке, *Xamarin.Android*, *Xamarin.iOS* и *Xamarin.Forms*, развијене су помоћу *Mono* платформе. *Mono* платформа представља отворену (енг. *open-source*) верзију *.NET* [5] окружења које може да се покрене на различитим оперативним системима (*Linux*, *Unix* и *macOS*).

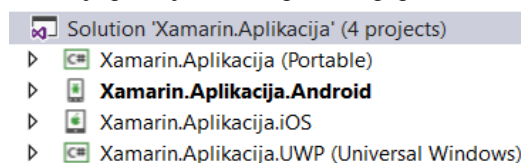
*Xamarin.Forms* је развијен на основу *Xamarin.Android* и *Xamarin.iOS* библиотека. Састоји се од *Xamarin.Forms.Core* библиотеке у којој се налазе класе у којима је дефинисан јединствени *API* за рад са више платформи. Садржи велики број функционалности које су мапиране на одговарајуће матичне функционалности појединачних платформи.

**2.2 Развој Xamarin.Forms апликација**

За развој *Xamarin* апликација могу да се користе *Visual Studio* [6] и *Xamarin Studio* [7] развојна окружења доступна за *Windows* оперативни систем, док се за *OS X* [8] оперативни систем користе *Xamarin Studio* и *XCode* [9]. *Xamarin* сет алата може да се преузме помоћу *Visual Studio* развојног окружења.

**2.2.1 Структура Xamarin.Forms пројекта**

Могућност развоја матичних апликација за више платформи захтева специфичну структуру пројекта која се састоји од пројекта везаних за сваку платформу и пројекта у коме се налази заједнички код за све платформе (енг. *shared code*). На слици 2.1 је приказан *Solution* са пројектима потребним да се апликација развија за све три платформе.

Слика 2.1. *Xamarin.Forms* solution

Пројекат означен на слици 2.1 као *Portable* је пројекат у коме се налази заједнички код за све три платформе, док су остали пројекти везани за специфичну платформу.

**2.3 Елементи Xamarin.Forms апликације****2.3.1 XAML**

Кориснички интерфејс (*UI*) у *Xamarin.Forms* апликацији може да се дефинише процедурално у класама и декларативно користећи *XAML* (*eXtensible Application Markup Language*). *XAML* није неопходан за дефинисање корисничког интерфејса, али је у већини случајева мање обиман и прегледнији од

эквивалентног C# кода. Користи хијерархију родитељ - дете што доприноси бољем представљању структуре корисничког интерфејса. XAML је посебно погодан уколико се користи MVVM (Mode-View-ViewModel) архитектура јер у XAML језику могу да се дефинишу погледи (View) који се повезују са ViewModel-ом.

### 2.3.2 Стране

Стране у Xamarin.Forms апликацији представљају визуелне елементе који се приказују на екрану мобилног уређаја. Стране су дефинисане Page класом и садрже погледе (тј. компоненте корисничког интерфејса помоћу којих корисници врше интеракцију са мобилном апликацијом). Постоји више унапред дефинисаних типова страница: ContentPage, NavigationPage, MasterDetailPage и TabbedPage.

### 2.3.3 Погледи (views)

Погледи представљају визуелне компоненте које се приказују на странама. Xamarin.Forms садржи више унапред дефинисаних погледа који се на мобилном уређају приказују као матичне компоненте.

### 2.3.4 Организација садржаја на екрану

За организацију садржаја који се приказује на екрану користе се распореди (енг. layout). Распореди су визуелни елементи који могу да садрже погледе или друге распореди.

### 2.3.5 Application класа

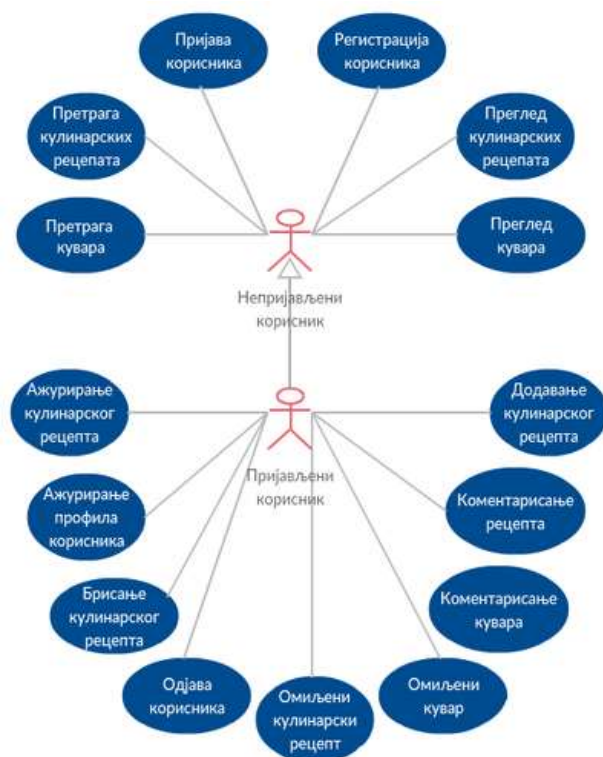
Application класа је основна класа Xamarin.Forms апликације. Представља страну која се прва отвара по покретању апликације, може да чува променљиве примитивног типа у пољу Properties које представља кључ-вредност структуру и садржи догађаје одговорне за приказивање и уклањање модалних погледа. У оквиру апликације је изложена App поткласом чији атрибут Current садржи референцу на тренутни објекат класе Application. Такође, садржи референцу на страну која се тренутно приказује (MainPage).

У Application класи су изложене методе животног циклуса апликације: OnStart (позива се током покретања апликације), OnSleep (позива се када апликација није више у првом плану) и OnResume (позива се када апликација поново дође у први план).

## 3. СПЕЦИФИКАЦИЈА ЗАХТЕВА

FoodBook мобилна апликација је имплементирана као пример употребе Xamarin.Forms алата. Апликација кориснику омогућава преглед, оцењивање и коментарисање кулинарских рецепата, преглед, оцењивање и коментарисање куvara, као и додавање сопствених кулинарских рецепата.

Све функционалности које корисник може да изврши на апликацији специфициране су коришћењем дијаграма случајева коришћења (енг. use case). На слици 3.1 приказан је дијаграм случајева коришћења FoodBook мобилне апликације.



Слика 3.1. Дијаграм случајева коришћења

## 4. СПЕЦИФИКАЦИЈА ДИЗАЈНА

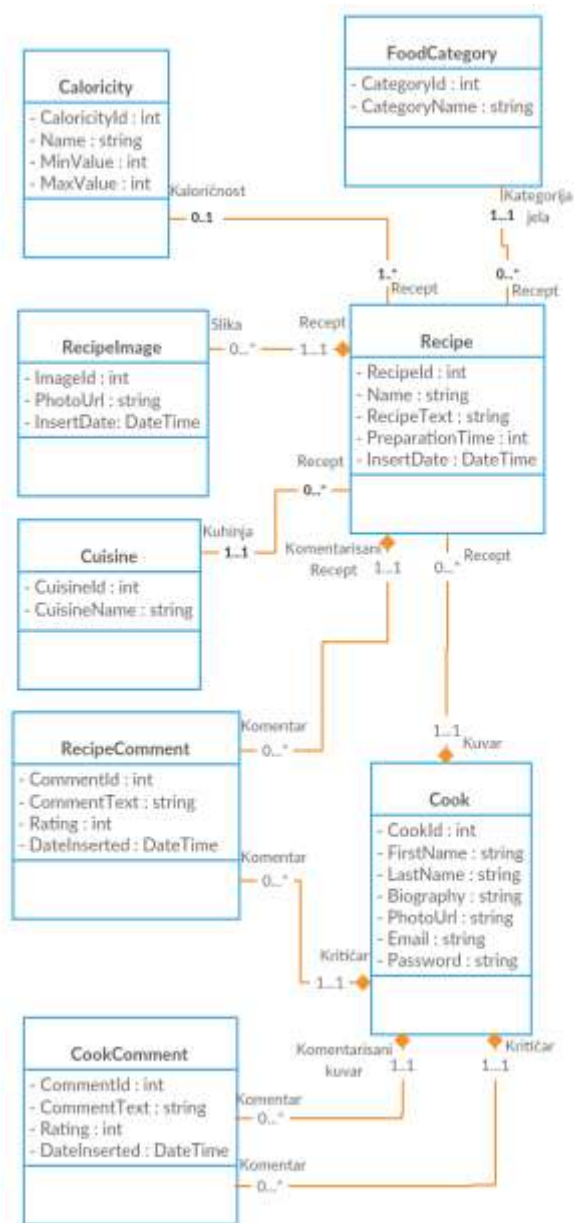
За опис модела података је употребљен дијаграм класа на ком су приказане класе и њихов међусобни однос.

На слици 4.1 приказане су класе и њихове међусобне везе. Класа Recipe представља кулинарски рецепт који може бити креиран од стране куvara који је представљен класом Cook.

Кулинарски рецепт припада одређеној категорији јела (класа FoodCategory), одређеној кухињи (класа Cuisine) и може садржати више слика (класа RecipeImage). Кулинарски рецепт може бити коментарисан од стране корисника мобилне апликације, коментар рецепта је представљен класом RecipeComment. Као и кулинарски рецепти, куvari такође могу бити коментарисани. Коментари куvara представљени су класом CookComment.

За приказ распореда компоненти апликације употребљен је дијаграм распореда. Дијаграм распореда FoodBook апликације приказан је на слици 4.2.

На дијаграму је приказано више компоненти од којих је главна компонента мобилна апликација имплементирана у Xamarin.Forms платформи. Мобилна апликација је састављена од компоненти у којима се налази заједничка пословна логика и компоненти везаних за конкретну платформу. Мобилна апликација комуницира са веб сервисом и на тај начин добавља и шаље податке.



Слика 4.1 - Дијаграм класа *FoodBook* апликације.

База података је компонента која складишти податке потребне мобилној апликацији. Поред базе података, веб сервис компонента комуницира са *Azure Cloud* [10] компонентом на којој се налазе компонента за складиштење слика и компонента за слање нотификација мобилној апликацији.

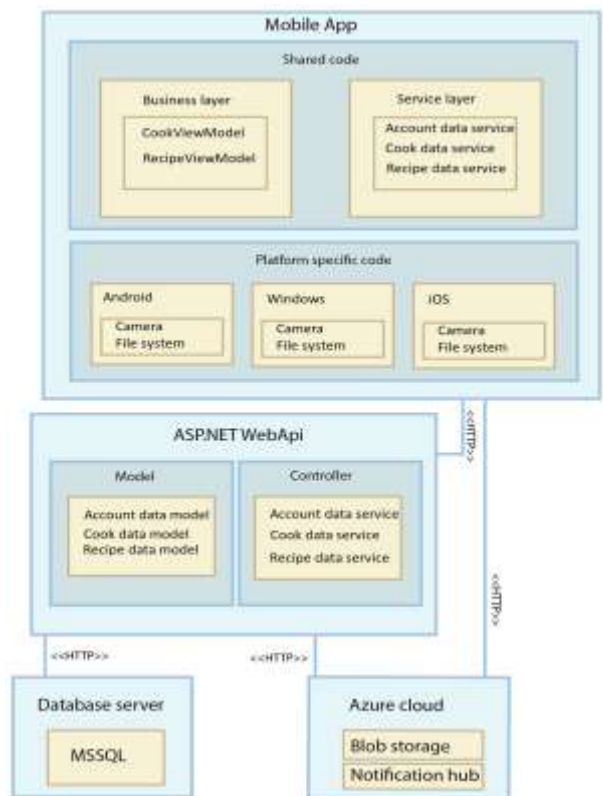
## 5. ИМПЛЕМЕНТАЦИЈА

Као пример употребе *Xamarin.Forms* платформе имплементирана је *FoodBook* мобилна апликација.

### 5.1 Почетак рада са платформом

За развој *FoodBook* мобилне апликације коришћено је *Visual Studio* развојно окружење на *Windows* рачунару што је довољна конфигурација за развој апликације за *Android* и *Windows* платформу.

Развој апликације за *iOS* платформу додатно захтева рачунар са *OS X* оперативним системом који се користи за извршавање кода *iOS* апликације.



Слика 4.2 - Дијаграм распореда *FoodBook* мобилне апликације

### 5.2 Имплементација екрана

За имплементацију страница апликације коришћен је *MVVM* архитектонски шаблон. Свака имплементирана страница садржи *Model*, *View* и *ViewModel* компоненте. Екрани *FoodBook* мобилне апликације приказују податке сачуване у бази података који се добављају позивајући веб сервисе. Након добављања података потребно је да се они припреме и проследи *View* компоненти која податке приказује на екран. Манипулација са подацима се обавља у *ViewModel* компоненти која садржи поља и команде које се везују за елементе корисничког интерфејса. Кориснички интерфејс, односно *View* компонента, садржи погледе помоћу којих се кориснику приказују одређени подаци.

Погледи помоћу којих корисник може да интерагује са апликацијом повезани су са пољима *ViewModel* компоненте техником везивања података (*data binding*).

### 5.3 Комуникација између компоненти

За комуникацију између независних компоненти апликације *Xamarin.Forms* нуди класу *MessagingCenter* која омогућава компонентама да се претплате на добијање информација које шаље нека друга компонента.

## 6. ДЕМОНСТРАЦИЈА

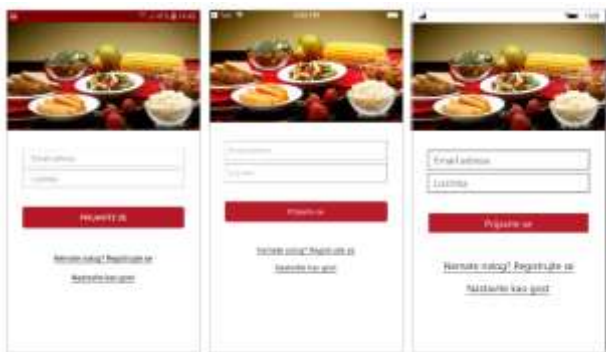
Након покретања апликације отвара се почетни екран за пријаву корисника (ако корисник није пријављен). Екран за пријаву корисника омогућава кориснику да се пријави на систем, а уколико није регистрован постоји опција за регистрацију корисничког налога.



Избор опције за регистрацију налога отвара нови екран на коме корисник уноси потребне податке за креирање новог налога. Након успешне регистрације корисник може да се пријави на апликацију уз помоћ креираног налога.

Ако корисник не жели да креира налог може приступити апликацији као непријављени корисник избором опције “Наставите као гост“. Непријављеном кориснику су доступни екрани за претраживање и преглед информација о постојећим кулинарским рецептима и куварима.

Пријављеном кориснику су такође доступни исти екрани али он има могућност креирања нових кулинарских рецепата, преглед детаља постојећих рецепата уз могућност избора кулинарских рецепата које су креирали остали корисници апликације у листу омиљених кулинарских рецепата. На екрану за преглед информација о рецепту пријављеном кориснику је доступна опција коментарисања тог рецепта, такође на екрану за преглед информација о одређеном кувару постоји могућност остављања коментара за тог куvara.



Слика 6.1. Екран за пријаву корисника приказан на Android, iOS и Windows платформама тим редоследом

## 7. ЗАКЉУЧАК

Тема овог рада је Xamarin.Forms оквир за имплементацију вишеплатформских мобилних апликација.

Xamarin.Forms је оквир вредан пажње зато што омогућава развој апликација за више платформи уз велики удео заједничког кода. Апликације развијене употребом ове платформе не заостају у великој мери у односу на апликације развијене коришћењем матичних технологија, а платформа омогућава коришћење једног програмског језика за имплементацију апликација на више платформи.

Поред свих добрих страна Xamarin.Forms платформе, током развоја и коришћења апликације приметне су и одређене мане. Не постоји стабилан прегледач корисничког интерфејса креираног у XAML документу, па креирани кориснички интерфејс може да се види тек након покретања апликације. Апликације креиране помоћу ове платформе заузимају доста више простора на мобилном уређају у односу на матичне апликације.

Иако је највећи део планираних функционалности имплементиран, FoodBook мобилна апликација може да се развија даље. Планиран је развој вишејезичности, као и додавање видео записа за кулинарски рецепт. Такође је планирана имплементација приватних профила корисника и приватних рецепата, којима могу да приступе само регистровани корисници. Рад апликације у режиму без интернет конекције би свакако био пожељан што би допринело да апликација буде доступна у сваком тренутку.

## 8. ЛИТЕРАТУРА

- [1] Android, <https://www.android.com/>
- [2] iOS, <http://www.apple.com/ios>
- [3] Windows, <https://www.microsoft.com/en-us/windows>
- [4] C#, <https://docs.microsoft.com/en-us/dotnet/csharp/>
- [5] .NET, <https://www.microsoft.com/net/>
- [6] Visual Studio, <https://visualstudio.microsoft.com/>
- [7] Xamarin Studio, <https://developer.xamarin.com/releases>
- [8] OS X, <https://support.apple.com/macOS/>
- [9] XCode, <https://developer.apple.com/xcode/>
- [10] Azure Cloud, <https://azure.microsoft.com/>

### Кратка биографија:

Милош Стевановић рођен је 26.05.1991. године у Приштини, Република Србија. У Смедереву је 2010. године завршио Средњу техничку школу, основне академске студије завршио је на Факултету техничких наука у Новом Саду 2014. године, а мастер академске студије на истом факултету уписао је 2014. године. Положио је све испите предвиђене студијским програмом.