



Универзитет „Св. Кирил и Методиј“ во Скопје  
**ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ И  
КОМПЈУТЕРСКО ИНЖЕНЕРСТВО**

## **Извештај за изработен проект**

Предмет: Напреден Веб Дизајн

**Тема: Flutter & Dart**



Dart  
Flutter

**Ментор:**

Проф. Д-р. Бобан Јоксимоски

**Изработил:**

Горги Лазарев

201042

# Содржина

Вовед во крос-платформски апликации и Flutter и Dart.....	3
Опис на развиената апликација.....	6
Развој на апликацијата.....	9
Технологии користени при развојот на апликацијата.....	9
Процес на развој на апликацијата .....	10
Заклучок и можности за дополнување на апликацијата .....	12

## Вовед во крос-платформски апликации и Flutter и Dart

Во денешен ден, апликациите се распространети насекаде и се дел од нашиот секојдневен живот. Најразлични апликации се користат за најразлични цели, од апликации за навигација и фитнес, до апликации за научно истражување и е-комерц, секој може да најде апликација која одговара на своите потреби и интереси. Апликациите најчесто ги подобруваат аспектите на нашиот живот и ја зголемуваат продуктивноста и можноста за комуникација, а тоа се должи на нивната распространетост на најразлични уреди, од компјутери и лаптопи до таблети, мобилни телефони и паметни часовници.

Крос-платформски апликации претставуваат одличен начин за развој на апликации кои можат да работат на повеќе оперативни системи, како Android, iOS, Windows, Linux, MacOS и Web-от со користење на ист код за сите платформи. Токму ова всушност го претставува и најголемиот бенефит од развивањето на крос-платформски апликации што тие се создаваат со еден ист код, со што се заштедува на време и ресурси и нудат ефикасен начин на развој на апликации. Покрај тоа, крос-платформските апликации овозможуваат поедноставен процес на одржување и надградба, бидејќи измените и поправките можат да се воведуваат на едно место и да се ажурираат на сите платформи автоматски.

Еден од иновативните начини за креирање на крос-платформски мобилни апликации е користењето на Flutter и Dart. Оваа рамка за развој на крос-платформски апликации стануваат се пошироко

застапени и претставуваат врвен инструмент во сферата на мобилните апликации и нудат многу можности за креирање на привлечни, брзи и функционални апликации.



Flutter е отворена рамка за развој на крос-платформски апликации, која е создадена од страна на Google. Таа овозможува креирање на крос-платформски апликации кои можат да работат на различни оперативни системи како што се Android и iOS, Windows, Linux и MacOS и Web-от со идентичен изглед и функционалност. Она што го прави Flutter особено атрактивен е неговата брзина и перформанси, што резултира во извонредно респонзивни апликации.

Со секоја нова верзија, Flutter се подобрува и збогатува со нови функционалности и можност за развој на апликации. Тоа го прави Flutter една од најзначајните технологии во областа на мобилниот развој и остава долготрајно влијание во индустријата.



Dart, пак, е програмскиот јазик врз кој што се базира Flutter и е исто така развиен и поддржуван од страна на Google. Овој јазик е

специјално дизајниран за развој на мобилни апликации со Flutter. Dart се одликува со својата ефикасност, брзина и лесна читливост, што го прави идеален за програмерите кои сакаат да создаваат квалитетни мобилни апликации.

Една од главните карактеристики на Dart е тоа дека е објектно-ориентиран јазик, што значи дека објектите и класите се основни компоненти во кодот. Ова им дава на програмерите можност да организираат и управуваат со кодот на подобар начин и да го направат повеќе структуриран и читлив.

Dart се истакнува и со својата ефикасност и брзина во извршувањето на кодот. Овој јазик користи just-in-time компајлер за развој и дебагирање на апликациите и автоматско преведување во машински код при извршување на апликацијата. Ова овозможува брзина и лесно дебагирање при развојот на апликации.

Со сите овие карактеристики и предности, Dart е јазик кој продолжува да расте во популарност и да се користи за развој на иновативни апликации во современите информатички индустрии.

Големиот број на достапни библиотеки и готови widget-и и компоненти за градење на кориснички интерфејси се главна причина, покрај можноста за развивање на крос-платформски апликации, поради која оваа рамка за развој на апликации, Flutter и програмскиот јазик Dart секојдневно добиваат нови корисници и поддршка.

Исто така, заедницата на корисници на Flutter и Dart е доста активна и нуди голема поддршка за секој што ја користи оваа рамка преку споделување на своите знаења и искуства.

Flutter и Dart, како и сите библиотеки и готови компоненти се многу добро документираны и содржат богати документации прикрупени со примери за лесна адаптација и користење во нашите апликации без да мора да бараме на милион места за да решиме некој проблем или имплементираме некое решение.

Како крај на воведот можеме да потенцираме дека Flutter и Dart се многу моќна алатка за развивање на крос-платформски апликации, а со револуцијата и се поголемата застапеност на паметните мобилни и

мали уреди, оваа алатка е многу привлечна и за оние кои прв пат се среќаваат со ваков тип на апликации и заедницата допрва ќе расте.

## **Опис на развиената апликација**

Апликацијата која што ја развив како дел од оваа проектна задача преставува апликација за наоѓање на превозници во рамките на Република Македонија. Платформата која што е развиена, всушност претставува две посебни апликации, една за клиентите, односно корисниците на кои им е потребен превоз и една за превозниците, односно провајдерите на услугата.

Апликацијата е инспирирана од Uber и Lyft, кои претставуваат најпопуларните апликации за барање на превоз и споделување на превоз во светот.

Апликацијата на клиентите им нуди платформа преку која можат да побараат превоз до било која дестинација во рамките на Македонија и во моментот на пишување можат да изберат од два типа на превозници, такси возило и приватно возило, при што при барањето и селекцијата на дестинацијата, автоматски се лоцира моменталната локација на корисникот и се пресметува трошокот за различните типови на превоз, како и оддалеченоста од дестинацијата и на корисникот му се дава можност за избор на тип на превозник и му се прикажуваат податоци како што се оддалеченоста, трошокот и му се исцртува целата рута која во тој момент е одберана како најкратка и најдобра на мапата во апликацијата. Откога корисникот ќе го избере посакуваниот тип на превоз, тој може да го поднесе барањето кое што ќе се пропагира до сите превозници од одбраниот тип кои се наоѓаат во радиус од 5 километри од моменталната локација на корисникот и при што се приоритизираат превозниците кои се најблиску до корисникот. Доколку има достапен превозник од посакуваниот тип на корисникот кој го прифатил барањето за превоз, тој се насочува кон корисникот и на корисникот му се прикажува во реално време статусот на патувањето, најпрво, додека да стигне превозникот до местото на клиентот му се прикажува колку минути е оддалечен според актуелната состојба на превозникот и сообраќајот каде што се движи, потоа кога превозникот ќе пристигне до корисникот и корисникот ќе го

започне патувањето, на корисникот му е прикажана рутата по која би требало да се движи превозникот и која е најдобра и најбрза во тој момент како и преостанатото време до пристигањето на дестинацијата. Откако корисникот ќе пристигне на дестинацијата и превозникот ќе го заврши патувањето, на корисникот му се прикажува приказ за износот кој треба да го плати за патувањето со платежната метода која ја избрал, во кеш или преку платежна картичка. По плаќањето, на корисникот му се прикажува екран за оценување на превозникот според искуството кое корисникот го поминал во текот на тоа патување. Ова е клучно за задржување само на добрите превозници и нивно приоритизирање при доделувањето на патувањата. Исто така на корисникот му е достапна можноста да го повика превозникот преку бројот кој што е запишан во системот и можност за откажување на патувањето пред да започне истото. Доколку пак во моментот на барање на превоз нема достапни превозници од бараниот тип, на корисникот му се прикажува порака дека може да побара друг тип на превозник или пак да почека одредено време додека да стане достапен превозник од бараниот тип, или пак доколку нема ниту еден достапен превозник, на корисникот му се прикажува порака да почека одредено време додека да се појави достапен превозник.

Од друга страна пак, апликацијата им овозможува на превозниците од различни типови да се регистрираат на платформата преку посебната апликација за превозници и да го зголемат обемот на работа а со тоа и можностите за профит со тоа што ќе бидат достапни кон сите корисници на апликацијата. Во моментот како превозници може да се регистрираат обични граѓани, кои имаат свој автомобил и сакаат да вршат превоз на патници со што ќе може дополнително да заработат или пак да ги споделат трошоците со некој доколку се движат на иста дестинација; возачи на такси возила, вработени во такси компании; локални и интернационални превозници на патници како и превозници на пакети – карго превозници. Иако сите овие типови на превозници може да се регистрираат во апликацијата, во моментот апликацијата им е достапна само на граѓаните кои сакаат да вршат превоз со сопствени возила и на возачите на такси возила, вработени во такси компании.

За секој тип на превозник се пополнуваат соодветните детали потребни за регистрација на тој конкретен тип на превозник.

Откако превозник ќе се регистрира и ќе ги пополни потребните податоци и ќе се најави, тој има можност да го промени својот статус како достапен или пак да го исклучи и да биде недостапен. При премин во состојба достапен, локацијата на превозникот се следи во реално време и се ажурира во системот и кај корисниците. Достапен превозник може да добие барање за превоз, доколку тој бил најблизок до одреден корисник кој побарал превоз на кое што му се прикажани локацијата на поаѓање, односно локацијата на корисникот кој бара превоз и посакуваната дестинација, при што може или да го одбие барањето или да го прифати. По прифаќањето на барањето, на превозникот му се исцртува рутата на мапата по која што тој треба да се движи за да најпрво пристигне до корисникот и му се прикажува предвиденото време до пристигање. Кога превозникот ќе стигне на местото за поаѓање, треба да го ажурира статусот на патувањето преку клик на копче кое е достапно кај превозникот. Откога превозникот и корисникот ќе го започнат патувањето, превозникот соодветно го ажурира статусот на патувањето и му се исцртува нова рута на мапата, која сега е од местото на поаѓање на корисникот до посакуваната дестинација. По пристигањето на посакуваната дестинација превозникот го завршува патувањето и му се прикажува износот на средства кој ќе го добие за тоа патување. По завршувањето на патувањето превозникот може да остане достапен за превоз или пак да го промени својот статус во недостапен.

На превозниците им се достапни неколку погледи преку пристап во менито и тоа: главниот поглед, каде што го ажурира својот статус на достапност и ја гледа мапата и му се исцртуваат рутите и појавуваат известувањата, потоа поглед каде што може да ги гледа дотогашните заработки и историја на секое патување кое го има направено преку апликацијата, со приказ на местото на поаѓање, дестинацијата, датумот на кој е завршено патувањето како и износот на средства кој го добил за тоа патување; поглед во кој може да ги гледа своите оценки, на скала од 1 до 5 при што се прикажува текстуален опис на оцените, од Многу лошо до Одлично и на крај поглед на кој се прикажани самите податоци за превозникот, како што се неговото име,



телефонски број, емаил и модел на автомобилот со кој врши превоз на патници.

## **Развој на апликацијата**

### **Технологии користени при развојот на апликацијата**

Останати технологии кои беа користени при развојот на апликацијата, покрај Flutter и Dart беа:

1. Firebase - развојна платформа позната по својата база на податоци во реално време, која претставува NoSQL база на податоци оптимизирана за синхронизирање податоци, често помеѓу кориснички машини или паметни телефони и централизирано складирање во облакот. Firebase исто така претставува Backend-as-a-Service, односно им овозможува на програмерите да го користат Firebase како бекенд за нивните апликации наместо да развиваат свој, со што заштедуваат на време и ресурси.  
Во рамките на оваа апликација, Firebase се користеше како бекенд, за автентикација на корисници, како база на податоци, преку користење на двата типа кои што ги нуди Firebase, Realtime Database – најмногу користена во оваа апликација поради можностите кои ги нуди во реално време и Firestore – за складирање на податоците за достапните превозници за Веб апликацијата; како провајдер за праќање на нотификации и пораки до достапните превозници преку Cloud Messaging.
2. FlutterFlow – алатка за визуелен развој на апликации која го олеснува креирањето на погледите преку овозможување на drag-and-drop едитори и готови компоненти. Оваа алатка служеше да изработка на некои од погледите во апликацијата и беше од големо олеснување во градењето на погледите преку нудење на готови имплементации на библиотеки за стилизирање на компонентите и widget-ите и utilities.

3. Google Maps API – API од страна на Google кое овозможува лесна интеграција на Google Maps во апликации и прилагодување на потребите на апликацијата. Голем дел од Google Maps API множеството беше искористен во развојот на апликацијата, пред се Maps API за приказ на самата мапа, Directions API за одредување и прикажување на рутите, Places API, за претварање на адреси во геометриски локации и обратно, како и за autocomplete при барање на дестинации.

## **Процес на развој на апликацијата**

Најпрво, апликацијата беше регистрирана со Firebase, за да може да се користи Firebase како бекенд. Соодветно, посебно беа регистрирани Android и iOS апликациите и посебно беа регистрирани Веб апликациите.

Потоа беше додадена FlutterFlow библиотеката за побрз развој на корисничките интерфејси и беше овозможена автентикација на корисници преку Firebase.

Потоа беше креиран Google Cloud проект интегриран со Firebase за овозможување на Google Maps API во апликацијата. Сите функционалности поврзани со приказот на мапата, и соодветните маркери, како и одредување на рутите, оддалеченоста и предвиденото време на пристигање, autocomplete при пребарување на дестинација беа имплементирани преку користење на подмножество од Google Maps API множеството и тоа: Maps API, Destinations API, Places API, Geocoding API.

За одредување на тековната локација беше користена библиотеката Geolocator.

Достапната локација како и сите променливи податоци кои треба да бидат перзистирани беа зачувани преку посебен DataHolder – AppData која претставува custom класа со потребните податоци за апликацијата и се перзистираа тие податоци преку употреба на Provider, кој ни овозможува промена и читање на податоци низ контекстот на целата апликација.

Потоа беше развиван корисничкиот интерфејс за корисниците, каде што некои од widget-ите беа креирани со помош на FlutterFlow.

API клучевите беа скриени со помош на environment варијабли соодветно и заштитени, односно ограничени за користење со цел да се спречи можна злоупотреба доколку дојде до несакаен приказ на тие клучеви.

Потоа беше имплементирано барањето на превоз преку зачувување на потребните податоци за патувањето во realtime база на податоци и потоа читање на тие податоци за испраќање на порака до достапните превозници.

Следењето на позицијата на достапните превозници во реално време беше направено со помош на Geofire библиотеката која отвара stream на кој што слуша за било какви промени во локацијата и таа локација ја ажурира во realtime базата на податоци. Бидејќи не постои имплементација на оваа библиотека за Веб, geoflutterfire2 библиотеката се користеше за соодветно овозможување на следење на позицијата на превозниците од страна на клиентите преку зачувување на локацијата во firestore базата и слушање на промените на документите во таа база.

Потоа беше имплементирано испраќањето на самото барање за превоз до најблискиот соодветен превозник преку Firebase Cloud Messaging, преку праќање на push нотификација до токенот на соодветниот превозник добиен од firebase и преку слушање на апликацијата за превозници на нотификации на тој конкретен токен.

При крајот беа имплементирани функционалностите за апликацијата за превозници, системот за оценување беше имплементиран со помош на библиотеката smooth\_star\_rating\_null\_safe, а приказот на историјата на патувања и заработките беше имплементиран преку читање на базата на податоци и ажурирање на променливите во AppData преку Provider-от.

За крај беа направени неколку козметички промени со кои се овозможува responsive дизајн на најразлични големини на уреди.

## **Заклучок и можности за дополнување на апликацијата**

Flutter и Dart се моќни алатки за развој на апликации кои претставуваат иновација во светот на програмирањето. Со Flutter, програмерите добиваат крос-платформска рамка за развој на апликации кои се одликуваат со одлични перформанси и современ дизајн, а пак Dart, како програмски јазик, им обезбедува ефикасност и лесна читливост на кодот. Заедно, Flutter и Dart им нудат на програмерите моќни алатки за креирање на апликации кои можат да ги задоволат потребите на модерните корисници и кои имаат можност да создаваат иновативни и функционални апликации кои работат на повеќе платформи и им овозможуваат на корисниците да се поврзат и интеракционираат со содржината на еден брз и ефикасен начин. Flutter и Dart продолжуваат да растат во популарност и да оставуваат значителен влијание во областа на мобилниот развој, и се очекуваат да продолжат да иновираат и внесуваат новости во иднината.

Додека пак, платформа развиена како дел од оваа проектна задача може да биде проширена со повеќе функционалности и да биде користена не само како платформа за превоз, туку како и платформа за изведување туристички екскурзии и организирани патувања, да им овозможува на корисниците брза и лесна комуникација со карго компаниите за достава на добра и можности за проширување на апликациите надвор од границите на Република Македонија.