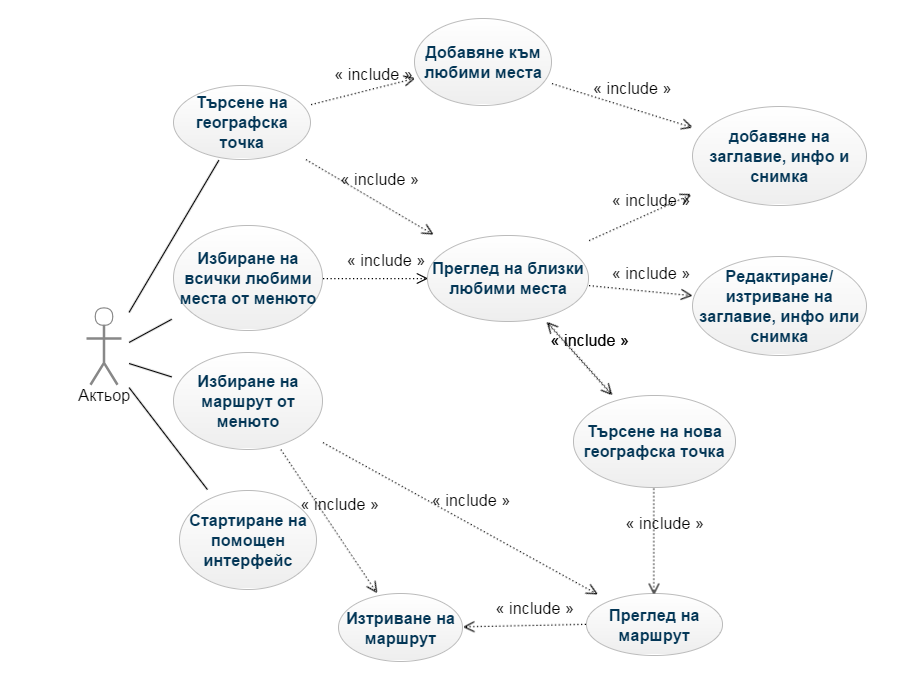
* Генерални изисквания:
  + след всяка извършена операция, трябва да се показва съобщение на потребителя – за успех, информативно или за грешка;
  + трябва да се имплементира помощен интерфейс, който да води потребителя през основните функционалности стъпка по стъпка;

### Диаграма на потребителските случаи (Use case diagram)

В процеса на проектиране на софтуерния продукт, диаграмата на потребителските случаи е първата диаграма, която се създава от проектантите, когато се започне проект. Тази диаграма позволява да се опишат на най-високо ниво целите на потребителя, които системата трябва да изпълнява. Тези цели не е необходимо да са задачи или действия, а може да са по общи изисквания към функционалността на системата. С други думи това е техника за определяне на функционалните изисквания на една система. Те описват типичните взаимодействия между потребителите и системата, предоставят описание на начина, по който тя се използва.

Компоненти при Use case диаграми, използвани в Trip Planner:

* Actor (Актьор) – роля, която един потребител играе по отношение на системата. Може да бъде клиент, представител по поддръжка на клиенти, мениджър продажби, продуктов аналитик. Актьорите изпълняват случаите на употреба. Един актьор може да изпълнява много случаи на употреба и един случай на употреба може да се изпълнява от много актьори. Актьора клиент може да представлява множество хора. Един човек може да играе ролята на няколко актьора. Актьора може да не е човек, а друга компютърна система, ако системата извършва услуга за нея.
* Associations – плътна връзка без стрелка. Аssociations са връзка между actors и use cases и означава, че актьора осъществява потребителския случай.
* Includes – Връзка показваща, че един Use sace включва друг Use case. Примера илюстрира как и покупката с фактура и покупката онлайн се включват в сценария описан от покупателната оценка. В обобщение, includes връзката е да отстрани повторенията на сценария в съставните случаи. Може да се каже, че includes е като извикване на процедура.

диагр. 1. Диаграма на потребителските случаи в Тrip Planner

Примерите за използване (Use-cases) са основани на сценарий техника в UML, която идентифицира актьорите в едно взаимодействие и която описва самото взаимодействие. Множество от примери на използване би трябвало да опише всички взаимодействия със системата.

Use case е съставен от множество от сценарии. Сценариите са примери от реалния живот, как системата може да се използва.

Следват текстови описания на някои от потребителските случаи (сценарии).

* Добавяне на ново любимо място:

|  |  |
| --- | --- |
| Потребителски случай | Добавяне на ново любимо място |
| Главен актьор | Потребител |
| Цел | Да се добави ново любимо място |
| Условия | Актьорът е стартирал уебсайта. |
| Причина | Необходимост от добавяне на нови любими места |
| Главен сценарий | * Актьорът въвежда адрес в полето за търсене. На дадената локация се добавя маркер; * Актьорът кликва върху маркера. Отваря се прозорче с бутон – Add to my places; * Актьорът кликва върху бутона. Показва се съобщение за успешно добавено ново място и на същата позиция се показва нов маркер за мястото; |
| Алтернативни сценарии | Няма |
| Изключения | * Не е намерено място – върху картата не се променя нищо. Потребителят има възможност да търси друго място; * Вече има добавено място на дадената локация – ново място не се добавя и се показва съобщение за грешка; |

* Редактиране на снимката на любимо място:

|  |  |
| --- | --- |
| Потребителски случай | Редактиране на снимката на любимо място |
| Главен актьор | Потребител |
| Цел | Да се смени снимката на любимо място |
| Условия | * Актьорът е стартирал уебсайта. * Има визуализирано любимо място върху картата. |
| Причина | Да се смени иконата на маркера на любимото място |
| Главен сценарий | * Актьорът кликва върху маркера. Отваря се прозорче с името на мястото и бутон – More info; * Актьорът кликва върху бутона. Отваря се по-голям прозорец, съдържащ полета за име, информация и снимка; * Актьорът кликва върху иконата за редактиране в долния десен ъгъл на прозореца. Активира се възможността за редакция на полетата; * Актьорът намира подходяща снимка в интернет и копира нейния адрес; * Актьорът поставя адреса на снимката в полето – Marker icon. Снимката се визуализира незабавно в долната част на прозореца. При затваряне на прозореца се вижда, че иконата на маркера също е сменена; |
| Алтернативни сценарии | Няма |
| Изключения | Адресът на снимката е невалиден url адрес – остава предходната маркер икона; |

* Построяване на маршрут:

|  |  |
| --- | --- |
| Потребителски случай | Построяване на маршрут |
| Главен актьор | Потребител |
| Цел | Да се построи маршрут между две или повече точки |
| Условия | Актьорът е стартирал уебсайта. |
| Причина | Да се визуализират близките любими места |
| Главен сценарий | * Актьорът въвежда адрес в полето за търсене. На дадената локация се добавя маркер и се визуализират близките до нея места; * Актьорът въвежда втори адрес в полето за търсене. На втората локация също се добавя маркер. Също така се визуализира маршрут между двете точки и любимите места, които са отдалечени не повече от предварително въведено максимално разстояние; |
| Алтернативни сценарии | При всяко следващо въвеждане на адрес в полето за търсене, се построява маршрут между новата и предходно намерената локация. |
| Изключения | Не е намерено място – върху картата не се променя нищо. Потребителят има възможност да търси друго място; |

## Нефункционални изисквания

Нефункционални изисквания дефинират ограничения върху системата или процеса на разработка. Те произтичат от нуждите на клиента, от политиките на организацията, от нуждата от взаимодействие с вече съществуващи системи или от външни фактори като правилници и закони.

Нефункционални изисквания се групират в следните категории:

* изисквания към продукта - определят поведението на продукта (надеждност, производителност, ефективност и други);
* организационни изисквания - произтичат от правила и политики в организацията-поръчител и организацията-изпълнител;
* външни изисквания - изисквания за съвместимост, закони, наредби и други;

### Използваемост

Използваемостта се отнася до качеството на потребителското преживяване при взаимодействие с продукти или системи, включително уеб сайтове, софтуер, устройства или приложения. Използваемостта се характеризира с ефикасност, ефективност и цялостно удовлетворение на потребителя. Използваемостта е комбинация от фактори, включващи:

* Интуитивен дизайн: почти без усилие разбиране на архитектурата и навигацията на сайта
* Лекота на учене: колко бързо един потребител, който никога не е виждал потребителския интерфейс, може да изпълни основни задачи
* Ефективност на използването: Колко бързо опитен потребител може да изпълнява задачи
* Запомняне: след посещение на сайта, ако потребителят може да си спомни достатъчно, за да го използва ефективно при бъдещи посещения
* Честота и тежест на грешката: колко често потребителите правят грешки при използването на системата, колко сериозни са грешките и как се възстановяват потребителите от грешките
* Субективно удовлетворение: Ако потребителят обича да използва системата

Изискванията към използваемостта на уебсайт за добавяне на любими места и планиране на пътешествия са:

* системата трябва да е достатъчно проста и интуитивна, така че да не отнема много време за разучаване;
* добавянето на ново любимо място трябва да става максимално бързо и лесно;
* любимите места трябва да се изобразяват върху географска карта, като е обозначена точната им локация;
* за всяко любимо място трябва да могат да се добавят и редактират малко информация и снимка, но те трябва да се достъпват на друг екран, с цел да се избегне претрупване на информация върху картата;
* максималната отдалеченост на местата близки до даден маршрут трябва да може да се променя според желанията на потребителя;
* трябва да могат да се запазват построени маршрути и да се разглеждат след това;

### Изправност

Изправността се състои от три компонента:

* надеждност - устойчивостта към грешки (независещи от потребителя) на софтуерния продукт по време на работа;
* сигурност - степента на устойчивост и защита от вредни въздействия, които биха могли да навредят както на софтуерният продукт, така и на устройството;
* готовност - времето, през което софтуерният продукт е годен за използване;

Изискванията към изправността на уебсайт за запазване на места и планиране на пътешествия са:

* системата трябва да е достъпна за употреба по всяко време, за да пести време на потребителя;
* системата трябва да работи надеждно и при нормална употреба да няма грешки по време на използване на основните функции;
* системата трябва да е сигурна и използването ѝ да не довежда до неизправности в устройството;

### Модулност и изменяемост

Модулно програмиране е техника за проектиране на [софтуер](https://bg.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80), увеличаваща степента, до която софтуерът е съставен от отделни, взаимозаменяеми компоненти, наречени модули. Това се постига чрез разбиване на функциите на програмата на модули, всеки от които изпълнява само една функция и съдържа всичко необходимо за нейното изпълнение. С други думи модулът е подсистема, част от [компютърна програма](https://bg.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8E%D1%82%D1%8A%D1%80%D0%BD%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B0), която изпълнява конкретни действия. Понякога модулът се нарича още пакет, [библиотека](https://bg.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5)) и др. Модулът съдържа предимно по-малки единици (функции, класове и т.н.) в повечето случаи, написани на отделен [файл](https://bg.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB).

Изменяемостта е възможността, когато се наложи да се направи промяна в системата, това да става лесно и бързо, без това да доведе до загуба на функционалност и създаване на грешки и изключения.

Наличието на модули помага за по-бързо разбиране на архитектурата на приложението и подобряват поддръжката и улесняват изменяемостта.