**„Софийски Университет**

**Свети Климент Охридски“**

****Logo

Description automatically generated

**Проектиране на Човеко-Машинен Интерфейс 2021 г.**

**Софтуерен Продукт (име: Планер4е)**

**Автори:**

**Гергана Георгиева, 62416; Тереза Димитрова, 62424; Ния Миковска, 62425**

**Велислава Петкова, 62428; Никола Ласков, 62429**

1. **Анализ на потребностите**
   1. **Основни цели и задачи**

Системата цели да помогне на потребителя за по-добра организация на времето, като му предложи нещо по-различно от стандартните приложения с „TO DO“ списъци и календари от типа на Google Calendar.

Целта ни е системата да обединява в себе си няколко различни приложения, които иначе потребителят би имал на телефона си, като така да му предостави повече удобство и всичко, което запише да е на едно място, вместо да се рови чудейки се къде е записал дадена задача.

* 1. **Целева група**
* Учащи млади хора

Това са хора, които имат нужда на едно място да виждат своите крайни срокове (домашни, тестове, изпити, проекти и т.н) и планирани събития. Развлекателните категории като Филми, Книги и Пътувания са особено подходящи за тази целева група, тъй като обхващат голяма част от главните интереси на подрастващите. Откъм гледна точка на дизайна, младото поколение харесва по-екстравагантни и разчупени екрани с по-ярки или пастелени цветове, а не неутрални такива. Разработването на системата като мобилно приложение е от съществена важност понеже предпочитаното от тях устройсво е мобилният телефон.

* Работещи хора

В тази целева група попадат хора с малко свободно време за планинг и менажиране на времето си и много задачи за изпълняване. Интерфейсът трябва да е лесен и интуитивен за използване и ориентиране в приложението. Съвкупността на няколко функционалности и видове списъци в едно е удобна откъм пестене на време за разучаване и използване на отделни приложения.

* 1. **Анализ на потребностите**
* **Проблеми, които ще решава системата**
  + Потребителите ще могат да организират своите задачи на едно място.
  + Потребителите ще могат да следят всичките си ангажименти.
  + Потребителите ще могат да получават известия за своите ангажименти.
* **Възможности, които предоставя системата**

Системата предоставя на потребителя възможност да подреди своите цели, задачи и интереси. Предоставят се различни предварително подготвени категории задачи, които потребителят може да попълва при желание, като освен това може да добавя разнообразни задачки за изпълнение в календара си и да им добавя допълнителни категории. Системата цели да се избегне изтеглянето на десетки приложения, всяко с отделна функционалност и да предостави, изчистен интерфейс, който да е удобен дори за потребител, който не използва всички функционалности.

* 1. **Изисквания към системата**

**Функционали изисквания:**

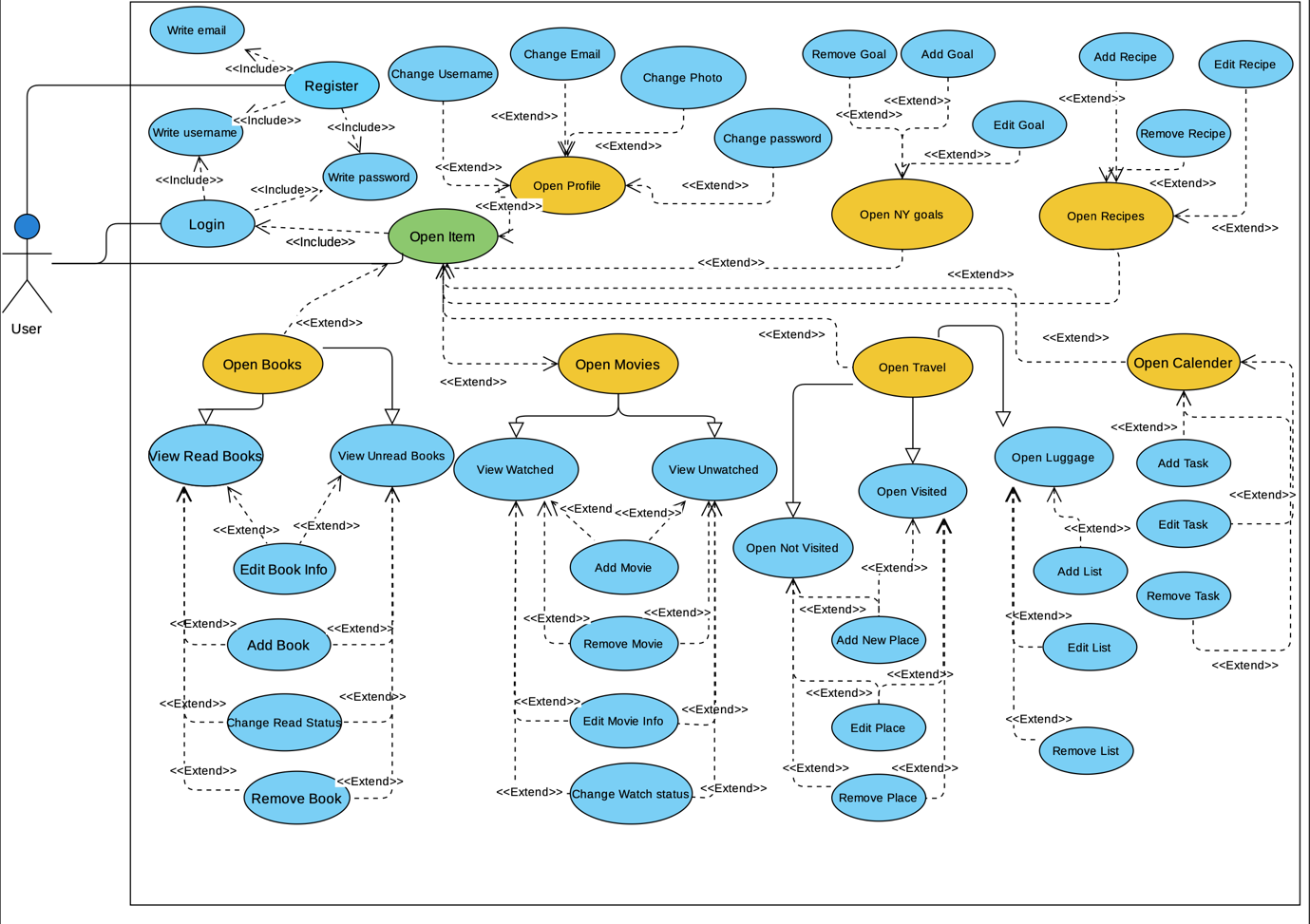
* Потребителят ще може да се регистрира, за целта ще трябва да се въведат имейл, име и парола.
* Потребителят ще може да влиза в системата със своите имейл и парола.
* Потребителят ще може да избира от меню към коя част на системата да се насочи: Календар, Филми, Книги, Новогодишни цели, Пътуване, Рецепти, Профил
* В Календар потребителят ще може да добавя нови персонализирани списъци със задачи, различни от гореспоменатите, като за всяка трябва да въведе Име на задачата, Дата на която или до която да се изпълни задачата, Една от трите възможни стойности за приоритет Low, Medium, High , Една от възможните категории Home, Work, Hobby, Shopping, Others, По желание на потребителя може да остави и допълнително описание под формата на коментар
* В Календар потребителят ще може да избере дали изгледа му да бъде за задачите за един ден или на календар за целия месец. При този изглед се вижда името на задачата, а приоритетът се изобразява с цвят Low - зелено, Medium – жълто и High – червено, а категориите Work, Hobby, Free time, Home, Others са иконки пред задачите.
* В Календар потребителят ще може да променя статуса на задача като изпълнена или неизпълнена чрез кликане на бутон в края на реда със задачата. Когато е завършена, задачата променя фона си към по-блед и се измества в края на страницата.
* В Календар потребителят ще може да преглежда по-подробна информация за дадена задача като кликне върхи нея. Информацията ще се показва на нов екран.
* В Календар потребителят ще може да редактира информацията за избрана задача след като я е отворил в нов екран.
* В Календар потребителят ще може да изтрие избрана задача, когато я отвори в нов екран.
* Потребителят ще може да добави нов филм към списъка с филми за гледане, като за всеки трябва да въведе, Име на филма, Приоритет на филма, По желание на потребителя може да остави и допълнително описание под формата на коментар
* Във Филми потребителят ще може да избира между списък с гледани филми и списък с филми за гледане.
* Във Филми потребителят ще може да променя статута на филм на гледан или за гледане.
* Във Филми потребителят ще може да преглежда подробна информация за даден филм, като кликне въру него. В нов прозорец се показва информацията, която е въвел като име, жанр и коментари.
* Във Филми потребителят ще може да променя информацията за избран филм, като кликне върху името на филма и се отвори подробната информация за него.
* Във Филми потребителят ще може да изтрива избран филм.
* Потребителят ще може да добави нова книга към списъка с книги за четене, като за всяка трябва да въведе Име на книгата, Приоритет на книгата, По желание на потребителят може да остави и допълнително впечатление под формата на коментар
* Потребителят ще може да избира между списък с четени книги и списък с книги за четене. В тези списъци се показват имената на книгите.
* Потребителят ще може да променя статута на книга на прочетена или непрочетена.
* Потребителят ще може да преглежда подробна информация за дадена книга, като кликне върху нея. Така ще се отвори прозорец с всичката въведена информация за книгата.
* Потребителят ще може да променя информацията за избрана книга като отвори прозорежа с подробна информация и редактира избраните полета.
* Потребителят ще може да изтрива избрана книга.
* Потребителят ще може да добави нова дестинация към списъка с дестинации за пътуване, като за всяка трябва да въведе Име на дестинацията, Приоритет на дестинацията По желание на потребителя може да остави и допълнителна информация под формата на коментар
* Потребителят ще може да избира между списък със посетени и не посетени дестинации, които съдържат имена на дестинациите, както и секция със списъци с багаж.
* Потребителят може да добавя нов списък с багаж, като му добавя име и неща, които трябва да вземе по време на път.
* Потребителят може да редактира даден списък като премахва и добавя елементи към него.
* Потребителят може да разглежда списък като кликне върху него, под списъка се показва съдържанието му.
* Потребителят може да променя име на списък с багаж.
* Потребителят може да изтрива списък, ако реши, че не му е необходим вече.
* Потребителят може да добавя и премахва тикчетата на всеки артикул в списъка с багаж, за да може да го преизползва.
* Потребителя ще може да променя статута на дестинация на посетена и не посетена.
* Потребителя ще може да преглежда подробна информация за избрана дестинация чрез кликане върху името ѝ и се отвори прозорец с цялата въведена информация за дестинацията.
* Потребителят ще може да променя информацията за избрана дестинация.
* Потребителят ще може да изтрива избрана дестинация
* Потребителят ще може да добави нова цели към списъка с неизпълнени цели, като за всеки трябва да въведе, Име на целта, Една от трите възможни стойности за приоритет Low, Medium, High, По желание на потребителя може да остави и допълнително описание под формата на коментар
* Потребителят ще може да избира между списък с неизпълнени и изпълнени цели, които съдържат съкратена информация за целите (име, дата и приоритет)
* Потребителят ще може да променя статута на цел на изпълнена или неизпълнена.
* Потребителят ще може да преглежда подробна информация за дадена цел като кликне върху нея и се отвори нов прозорец с въведената информация.
* Потребителят ще може да променя информацията за избрана цел като кликне върху нея и се отвори нов прозорец с въведената информация и там избере полетата, които иска да редактира.
* Потребителят ще може да изтрива избрана цел
* Потребителят ще може да добави нов рецепта, като за всяка трябва да въведе, Име на рецептата, Текст на рецептата, При желание може да добави и снимка към рецептата
* Потребителят може да преглежда списък от рецепти с кратка информация за тях (име и снимка)
* Потребителят може да преглежда подробна информация за дадена рецепта като кликне върху нея и се отвори цялата информация в нов прозорец.
* Потребителят ще може да променя информацията за избрана рецепта като кликне върху нея и се отвори цялата информация в нов прозорец. Там той може да избере и редактира желаните полета.
* Потребителят ще може да изтрива избрана рецепта
* Потребителят може да види името, имейла и профилната си снимка
* Потребителят може да качи профилна снимка
* Смени данните за профила си(Име, парола, имейл или снимка)
* Потребителят може да излезе от профила си
* Потребителя може да смени езика на приложението си

**Нефункционални изисквания:**

**Производителност:**

* Системата може да допуска до 10 000 нови потребителя на месец
* Системата може да поддържа до 100 потребителя едновременно
* Потребителят има достъп до системата по всяко време на деня, като изключим предварително обявени интервали за поддръжка
* Системата се поддържа на английски език
* Предвидени са 2 часа на месец за поддръжка на системата
* Не се дава достъп на неоторизирани лица до личните данни на потребителите
* Всеки потребител има достъп само до информация въведена от него във системата
* Личните данни на потребителите са обработвани спрямо регулациите на наредбата GDPR
* Платформата е достъпна 96% от времето
* Интерфейса е сходен с други подобни системи за да се осигури лесна и интуитивна работа с него
* Доставя се лесен достъп до всички функционалности на приложението
* Поддържа 2 езика(български и английски)

1. **Проектиране на системата**
   1. **Use Case Diagram**

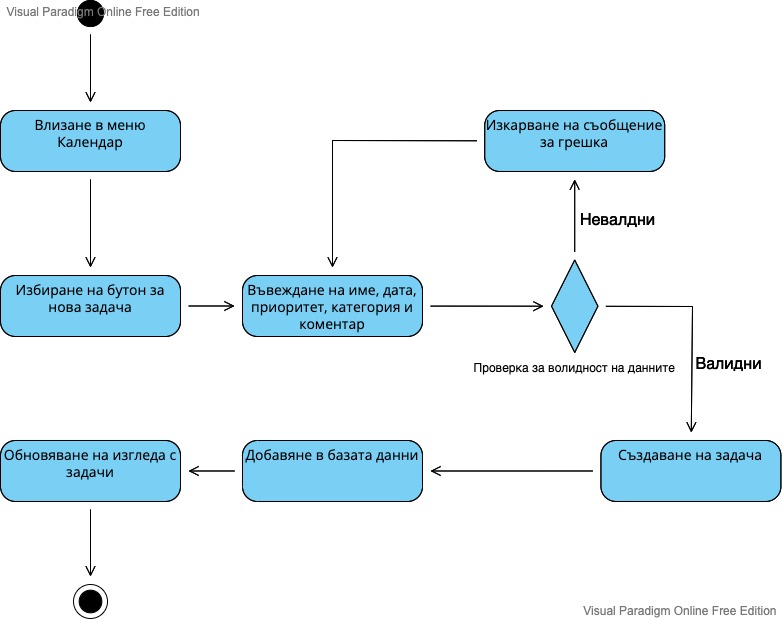
****

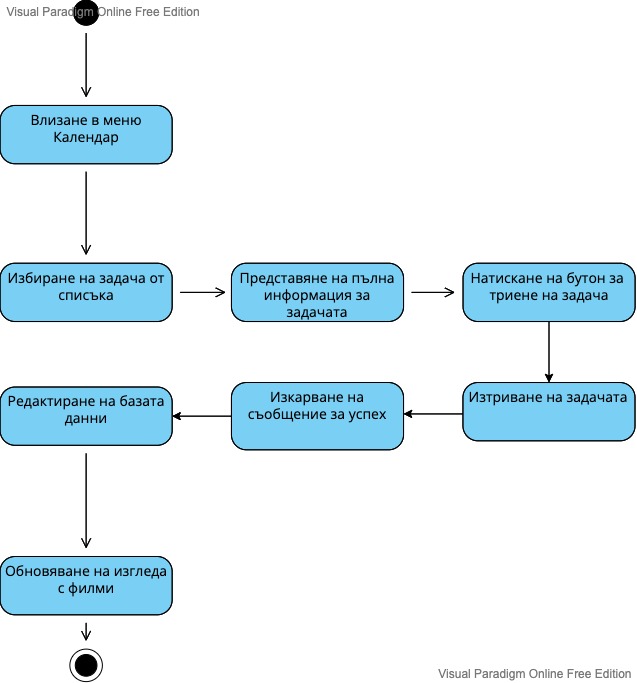
* 1. **Use Cases**

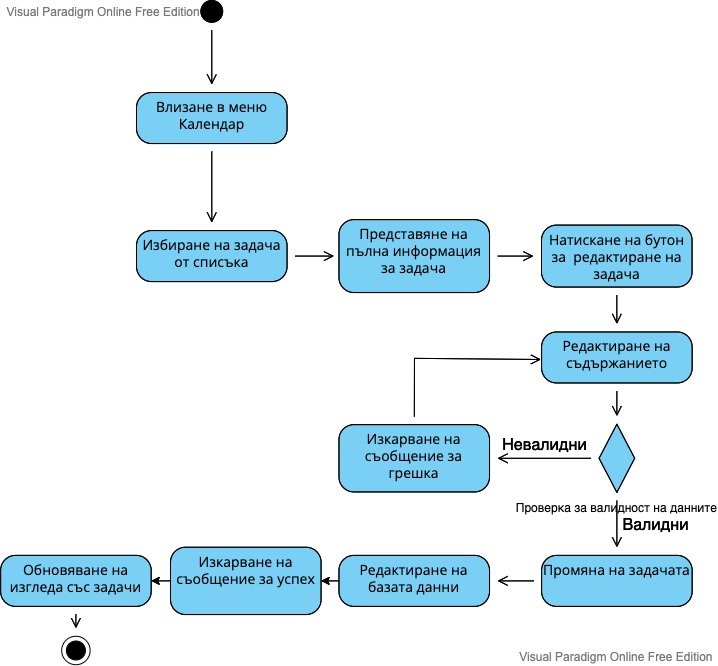
|  |  |  |
| --- | --- | --- |
| **Use Case ID:** | USE CASE:1 | |
| **Use Case Name:** | Добавяне на задача (Task) | |
| **Actors:** | | 1. Регистриран потребител 2. Система на базата данни |
| **Description:** | | Потребителя ще създаде нова задача, която да се появи във списъка със неизпълнени задачи. |
| **Preconditions:** | | 1. Потребителя трябва да има регистрация в приложението 2. Потребителя трябва да е влезнал в профила си. 3. Потребителя трябва е влезнал в прозореца за добавяне на нова задача. |
| **Postconditions:** | | 1. Във списъка със задачи на потребителя се появява нова задача. |
| **Normal Flow:** | | 1. Потребителя въвежда име на задачата 2. Потребителя избира приоритет на задачата (Нисък, Среден, Висок) 3. Потребителя избира категория на задачата (Дом, Работа, Хоби, Пазаруване, Други) 4. Потребителя избира дата за задачата 5. Потребителя избира дали да въведе коментар 6. Потребителя натиска бутона „Запази“ 7. Системата запазва новата задача в базата с данни 8. Приложението прехвърля на списъка със задачи, където може да се види новата задача |
| **Alternative Flows:** | | * 1. Ако лисва име на задачата се зарежда грешка с надпис „Трябва име.“   2. Ако лисва избрана категория се зарежда грешка с надпис „Трябва категория.“   3. Ако лисва избран приоритет се зарежда грешка с надпис „Трябва приоритет.“   4. Ако лисва избрана дата се зарежда грешка с надпис „Трябва дата.“ |
| **Notes and Issues:** | |  |

|  |  |  |
| --- | --- | --- |
| **Use Case ID:** | USE CASE:2 | |
| **Use Case Name:** | Редактиране на задача (Task) | |
| **Actors:** | | 1. Регистриран потребител 2. Система на базата данни |
| **Description:** | | Потребителя ще редактира вече добавена задача. |
| **Preconditions:** | | 1. Потребителя трябва да има регистрация в приложението 2. Потребителя трябва да е влезнал в профила си. 3. Потребителя трябва е влезнал в списъка със задачи. |
| **Postconditions:** | | 1. Във списъка със задачи на потребителя се появява редактирана задача |
| **Normal Flow:** | | 1. Потребителя избира задача 2. Приложението прехвърля към екран с подробна информация за задачата 3. Потребителя натиска бутона „Edit“ за редактиране на задачата 4. Потребителя редактира някое от полетата на задачата от задачата 5. Потребителя натиска бутона „Запази“ 6. Системата променя задачата в базата с данни 7. Системата показва съобщение за успех 8. Приложението прехвърля на списъка със задачи |
| **Alternative Flows:** | | * 1. Ако лисва име на задачата се зарежда грешка с надпис „Трябва име.“   2. Ако лисва избрана категория се зарежда грешка с надпис „Трябва категория.“   3. Ако лисва избран приоритет се зарежда грешка с надпис „Трябва приоритет.“   4. Ако лисва избрана дата се зарежда грешка с надпис „Трябва дата.“ |
| **Notes and Issues:** | |  |

|  |  |  |
| --- | --- | --- |
| **Use Case ID:** | USE CASE:3 | |
| **Use Case Name:** | Изтриване на задача (Task) | |
| **Actors:** | | 1. Регистриран потребител 2. Система на базата данни |
| **Description:** | | Потребителя ще изтрие съществуваща задача задача. |
| **Preconditions:** | | 1. Потребителя трябва да има регистрация в приложението 2. Потребителя трябва да е влезнал в профила си. 3. Потребителя трябва е влезнал в списъка със задачи. |
| **Postconditions:** | | 1. Във списъка със задачи на потребителя се появява обновен списък със задачи |
| **Normal Flow:** | | 1. Потребителя избира задача 2. Приложението прехвърля към екран с подробна информация за задачата 3. Потребителя натиска бутона „Delete“ за редактиране на задачата 4. Системата изтрива задачата от базата с данни 5. Системата показва съобщение за успех 6. Приложението прехвърля на списъка със задачи |
| **Alternative Flows:** | |  |
| **Notes and Issues:** | |  |

* 1. **Activity Diagrams**

****



1. **Описание на реализацията**
   1. **Избор и обосновка на избраните технологии**

За основен език на които е писано приложението избрахме SWIFT защото той е един от по-новите и бързо развиващи се езици за мобилни приложения. За база данни и авторизация на юзерите сме използвали услугата на Google FIREBASE(https://firebase.google.com/). Тя предоставя безплатна облачна услуга за учебни проекти и е достатъчно бърза и обхватна за нашите нужди. За сваляне на големи файлове от FIREBASE Storage сме използвали готова библиотека KingFisher (<https://cocoapods.org/pods/Kingfisher>) която позволява по-бърза работа и удобна работа със FIREBASE Storage. Библиотеката KingFisher е написана на Ruby.

* 1. **Описание на реализацията на основните функции.**

1. Първо се реализира под-система за регистрация и вход на потребители

2. Създаде се основа (Основно меню) към която се добави системата за вход и регистрация

3. Изгради се база данни в която да се пазят данни въвеждани от потребителите

4. Всяко меню беше първо реализирано като отделна под-система която по-късно беше интегрирана към основата:

a. Създаване на под-система Задачи

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

b. Създаване на под-система Книги

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

c. Създаване на под-система Филми

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

d. Създаване на под-система Цели

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

e. Създаване на под-система Рецепти

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

f. Създаване на под-система Профил

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

g. Създаване на под-система Пътуване

i. Създаване на UI за различните прозорци

ii. Добавяне на функционалност

5. Добавяне на възможност за втори език (английски)

6. Тестване с помощта на потребители

* 1. **Демонстриране на реализацията на графичния интерфейс и обосновка на основните решения**

За демонстрация може да разгледате прекаченото видео Demo.

Основната идея беше да се спазват дизайн принципите на IOS приложенията и да се осигури изчистен интерфейс. Искахме потребителят лесно да се ориентира в основното меню и да има отделна препратка към всеки вид списък, за да са отделени и да е по-удобно за потребителят ако не използва всички видове списъци. Постарахме се да запазим еднакъв формат на отделните категории за по-лесна ориентация и да предоставим календар, за да има по-лесно визуално възприятие на задачи и крайни срокове.

* 1. **Анализ и обосновка на избраните стратегии за навигация.**

Навигацията в приложението е постигната чрез основно меню, което е корен за всички възможности във приложението. За навигацията между екраните се използва вградения във SWIFT UIKIt обект “UINavigationController”, който изгражда дървовидна връзка между различните екрани. Този обект предоставя бутон в горния ляв ъгъл на екрана, който позволява връщане към предишни екрани. За прехвърляне между екрани които не са един до друг в дървовидната структура се използват “Segue” обекти, дефиниращи връзка между два екрана.

* 1. **Инсталация,поддръжка,потребителска документация**

Tъй като приложението е за мобилно устройство със IOS операционна система, за инсталиране на приложението на компютър, лаптоп или MacBook е необходима XCODE софтуера на Apple със включен “xCode IOS Simulator” (виртуален IPHONE). За да се зареди приложението се отваря файла The Time Project.xcworkspace и се избира приложението да се включи на симълатор на IPHONE 11. За инсталиране на мобилен телефон си иска допълнително разрешение от страна на потребителя, за това че вярва на този разработчик(различно за всеки телефон).

1. **Оценяване**

За тестване на приложението са създадени потребителски сценарии, които да бъдат представени пред бъдещи потребители и да се проследи техния успех със постигането на дадените задачи. Освен данни по време на тестването се събират и крайни мнения за целостта на приложението.

Пред бъдещи потребители бяха поставени следните цели:

1. Създаване на собствен профил
2. Смяна на езика
3. Добавяне на нова задача
4. Редактиране на добавена задача
5. Изтриване на задача
6. Добавяне на рецепта
7. Излизане от профила

Всеки потребител който ще тества приложението е представен със списък от цели и се записва опитът му да ги изпълни. След края на тестването се дава и крайно мнение под формата на текстов документ или чрез вербална комуникация

Потребителите се справиха със поставените цели, най-дълго време от които отнеха редактиране и изтриване на задача. По-младите потребители се справиха по-бързо от останалите и проявиха интерес и за бъдещи функционалности на приложението.

Крайните мнения бяха основно за семпъл, прост и разбираем интерфейс, с изключение на един потребител които смяташе че има твърде много функционалности.

Основна цел за развитието на софтуера ще е разработването на версии и за други устройства, като телефони с операционна система Android и Web версия, за да достигнем до повече потребители. За бъдещето развитие на приложението е възможно и прехвърляне към платена облачна услуга която да позволява по-голямо количество регистрирани потребители, по-бърза връзка, както и повече място. Също така е възможно подобряване на потребителския интерфейс и включване на възможност за смяна към по-опростен интерфейс с по-малко възможности за възрастните хора.