**CENTRO PAULA SOUZA**

**FACULDADE DE TECNOLOGIA DE FRANCA**

**“Dr. THOMAZ NOVELINO”**

**TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

João Paulo Fernandes rodrigues

**A TECNOLOGIA DIGITAL DA INFORMAÇÃO E COMUNICAÇÃO PARA MELHORIA DA QUALIDADE DE VIDA DE PESSOAS DIABÉTICAS**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Claudio Eduardo Paiva

FRANCA/SP

2022

**A TECNOLOGIA DIGITAL DE INFORMAÇÃO E COMUNICAÇÃO PARA MELHORIA DA QUALIDADE DE VIDA DE PESSOAS DIABÉTICAS**

**João Paulo Fernandes Rodrigues [[1]](#footnote-1)**

**Resumo**

O diabetes é uma doença que afeta o pâncreas, causando seu mal funcionamento na produção de insulina, que é responsável por controlar o nível de glicose. O mau controle dessa doença pode acarretar sérios danos à saúde, em alguns casos levando a pessoa a morte, sendo assim a proposta deste trabalho é apresentar o desenvolvimento de um software para auxiliar essas pessoas a realizarem um melhor controle e monitoramento de sua doença, de forma correta e eficaz. Softwares que auxiliam no controle de diabetes através de verificações do índice de glicemia, registros de glicemia, e monitoramento da doença podem ser de grande contribuição social, proporcionando uma melhor qualidade de vida a seus usuários. Para desenvolver este projeto foi-se utilizado do convívio com diversas pessoas portadoras desta doença, bem como estudos de métodos para se obter um melhor controle desta comorbidade, e estudos das principais ferramentas de elicitação de requisitos e de recursos mais utilizados para o desenvolvimento de software para ambientes multiplataformas. Utilizou-se do framework Xamarin Forms que foi responsável por fazer a interligação da interface com o banco de dados SQLite, utilizando a linguagem de programação C#. Este projeto resultou em um sistema que realiza cálculos complexos de correções de glicemia e faz agendamento de dados, permitindo assim monitorar a doença. Por tanto o software desenvolvido incentiva uma melhor qualidade de vida para pessoas portadoras de diabetes, proporcionando facilidades para se conseguir controlar o nível de glicemia do usuário, de forma rápida e segura.

**Palavras-chave:** Controle. Diabetes. Insulina. Monitoramento. Multiplataformas.

***Abstract***

Diabetes Mellitus is a disease that affects the pancreas, causing it to malfunction in the production of insulin that is responsible for controlling the glucose level. Poor control of this disease can cause serious damage to health, in some cases leading to death, so the purpose of this work is to present the development of software that is capable of helping these people to perform better control and monitoring of diseases. your illness, correctly and effectively. Software that helps control diabetes through glycemic index checks, glycemic records, and disease monitoring can be of great social contribution, providing a better quality of life for its users. To develop this project, it was used the contact with several people with this disease, as well as studies of methods to obtain a better control of this comorbidity, and studies of the main requirements elicitation tools and the most used resources for software development. for cross-platform environments. The Xamarin Forms framework was used, which was responsible for interconnecting the interface with the SQLite database, using the C# programming language. This project resulted in a system that is capable of performing complex calculations of blood glucose corrections, data scheduling, and disease monitoring. Therefore, the developed software encourages a better quality of life for people with diabetes, providing facilities to be able to control the user's blood glucose level, quickly and safely.

***Keywords:*** *Control. Diabetes. Insulin. Cross-platform.*

1 Introdução

O diabetes é uma doença que afeta primordialmente o pâncreas do indivíduo, sendo que este órgão é responsável pela produção de insulina, o mau controle da doença pode gerar danos irreversíveis a saúde da pessoa, e em alguns casos levando a morte, sendo assim, este trabalho tem por finalidade apresentar o desenvolvimento de um sistema multiplaraformas que seja capaz de proporcionar uma melhor qualidade de vida a seus usuários, através de sugestões de dosagens de insulina para que se possa ter um melhor controle e monitoramento sobre a doença.

A função da insulina é promover a entrada de glicose para as células do organismo de tal maneira que ela possa ser aproveitada para as diversas atividades celulares. A falta de insulina ou um defeito na sua ação promove acúmulos de glicose no sangue, esse acúmulo é chamado de hiperglicemia. É de extrema importância tratar da hiperglicemia, tendo em vista que está associada a lesões da microcirculação, o que ocasiona um mal funcionamento em diversos órgãos como rins, olhos, nervos e coração.

Atualmente, aproximadamente 415 milhões de adultos apresentam Diabetes Mellitus (DM) em todo o mundo e 318 milhões de adultos possuem intolerância à glicose, com risco elevado de desenvolver a doença no futuro.

O Brasil é o quarto país com maiores taxas de DM na população adulta, com um total de 14,3 (12,9-15,8) milhões de pessoas de 20 a 79 anos com DM, com um gasto anual estimado de pelo menos US$ 21,8 bilhões (BRASIL, 2017, p.09).

Os tipos de diabetes mais frequentes são o diabetes tipo 1, que corresponde a 10% do total dos casos, e o diabetes tipo 2, que corresponde a 90% dos casos (BRASIL, 2017, p.12).

O diabetes tipo 1, também conhecida como diabetes juvenil, indica a deficiência absoluta do pâncreas na produção de insulina. Nesse tipo específico de diabetes o uso injetável de insulina passa a ser obrigatório e de extrema importância para que se possa prevenir a cetoacidose que se trata de uma complicação da diabetes em que se pode tornar fatal devido à falta de controle da doença.

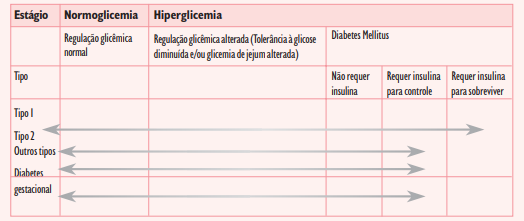
No caso da diabetes do tipo 1, o organismo é incapaz de produzir insulina, o que faz com que a glicose permaneça em elevadas concentrações no sangue e em baixa quantidade dentro das células. Isso faz com que o organismo passe a utilizar a gordura como fonte de energia para manter as funções do corpo, levando à produção de corpos cetônicos em excesso, o que recebe o nome de cetose.

A presença em excesso de corpos cetônicos faz com que haja diminuição do pH do sangue, deixando-o mais ácido, o que é chamado de acidose. Quanto mais ácido é o sangue, menor é a capacidade do corpo de realizar as suas funções, o que pode levar a pessoa ao coma e, até mesmo, ao óbito (BEZERRA, 2021).

O diabetes tipo 2 consiste no mal funcionamento do pâncreas que gera uma deficiência relativa à insulina. Em alguns casos do diabetes tipo 2 é necessário o uso de insulina, porém, ao contrário da diabetes tipo 1, o uso de insulina não visa evitar a cetoacidose, mais sim obter o controle de hiperglicemia, como pode-se ver na Figura 1. A cetoacidose nesse caso é rara, e quando presente geralmente está acompanhada de alguma infecção, trauma ou um estresse muito grave.

Outro tipo de diabetes que surge com mais frequência é o diabetes gestacional, que se identifica em mulheres no período pré-natal. Outras variações do diabetes menos frequentes surgem devido de defeitos genéticos da função das células beta, ocasionando sintomas da doença (BRASIL, 2017, p.12).

**Figura 1 –** Requisitos para uso de insulina



**Fonte:** Brasil (2016)

É possível identificar pessoas portadoras de diabetes mellitus tipo 1 e tipo 2 pelos seguintes sintomas: vontade de urinar com muita frequência, sede excessiva, fome em excesso, perda involuntária de peso, cansaço frequente, sonolência, coceira em todo corpo, infecções frequentes, irritabilidade e mudanças de humor repentinas.

Já as pessoas com diabetes tipo 2 geralmente apresentam os seguintes sintomas: aumento de sede, boca constantemente seca, vontade de urinar com muita frequência, cansaço frequente, visão turva ou embaçadas, feridas com a cicatrização muito lentas e infecções frequentes (MINISTÉRIO DA SAÚDE, 2009).

Para que uma pessoa portadora de diabetes mellitus possa ter uma melhor qualidade de vida elenca-se como essencial a alimentação e a prática de exercícios físicos que ajudam na distribuição de glicose entre as células. Na alimentação a quantidade energética ingerida deve ser adequada à atividade física e ser fracionada em 5 a 6 refeições/lanches diários. Para tanto, os pacientes devem ser encorajados a comer alimentos ricos em fibras, como frutas, verduras, legumes, feijões e cereais integrais. Com relação aos diabéticos tipo 1, obrigatoriamente, a aplicação de insulina é feita de acordo com a quantidade de carboidratos consumidos e deve ser definida por um médico.

A terapia nutricional é parte fundamental do plano terapêutico do diabético. Baseia-se nos mesmos princípios básicos de uma alimentação saudável. A quantidade energética ingerida deve ser adequada à atividade física e ser fracionada em 5 a 6 refeições/lanches diários. Portanto, a dieta deve ser individualizada, respeitando as características de cada indivíduo.

A ingestão diária deve conter carboidratos, a maior parte em forma complexa. Para tanto, os pacientes devem ser encorajados a comer alimentos ricos em fibras, como frutas, verduras, legumes, feijões e cereais integrais.

A ingestão diária de gordura deve ser baixa, principalmente as de origem animal (ácidos graxos saturados) (DALTOÉ, 2013).

As correções no nível de glicemia devem ser feitas com base em exames feitos em um aparelho chamado glicosímetro, em termos gerais o paciente coleta uma pequena quantidade de sangue retirada da ponta de seu dedo, colocando em uma pequena plaqueta onde será analisado e informado o valor atual de sua glicemia. A base para o cálculo de correção é definida obrigatoriamente pelo médico.

A glicemia capilar é medida por meio de uma pequena quantidade de sangue que é retirada da ponta do dedo e que é analisada pelo glicosímetro, que é o nome dado ao equipamento (BEZERRA, 2021).

O diabetes apresenta altos índices de internações e morbimortalidade, o que ocasiona uma perda importante na qualidade de vida do portador de tal doença, isso ocorre devido a erros ao fazer as correções através de insulina e mal monitoramento da doença (BRASIL, 2006). Destaca-se ainda que a doença pode ser um dos motivos de algumas infecções encontradas nos pacientes. O diabetes é uma das principais causas de mortalidade e pode causar insuficiência renal, amputações de membros inferiores, cegueira e doenças cardiovasculares.

A expectativa de vida é reduzida em média em 15 anos para o diabetes tipo 1 e em 5 a 7 anos na do tipo 2; os adultos com diabetes têm risco 2 a 4 vezes maior de doença cardiovascular e acidente vascular cerebral; é a causa mais comum de amputações de membros inferiores não traumática, cegueira irreversível e doença renal crônica terminal (BRASIL, 2006).

Estima-se que uma em cada 12 mortes em adultos no mundo possa ser atribuída ao DM, totalizando, aproximadamente, cinco milhões de casos ao ano, o que equivale a uma morte a cada seis segundos. A proporção de óbitos é ligeiramente maior em mulheres do que em homens. O gasto com DM, na maioria dos países, varia entre 5% e 20% das despesas globais em saúde (Brasília, 2017, p.09).

Diante disso, este trabalho tem o objetivo de apresentar o desenvolvimento de um software para controle e monitoramento dos níveis de glicemia de pacientes com diabetes e, assim, proporcionar uma melhor qualidade de vida meio do controle da quantidade de insulina a ser injetada no sangue utilizando-se de tecnologia da informação. Para isto, um aplicativo multiplataforma foi criado por meio da ferramenta de desenvolvimento Xamarin Forms a fim de facilitar a definição de horários e cálculos rápidos de alimentação a ser consumida, possibilitando o registro dos dados de medições para monitoramento do tratamento.

O aplicativo visa evitar erros de cálculos complexos referentes à quantidade de insulina a ser aplicada e também oferece funcionalidades como alertar o usuário para tomar seus remédios nos devidos horários e fazer exames de correções, registrar observações referentes aos horários de realização dos exames e das atividades físicas, lembrar sobre as próximas consultas e exames marcados, além de auxiliar o médico a obter um cronograma e monitoramento mais detalhado do paciente, proporcionando uma melhor qualidade de vida para pessoas portadoras do diabetes.

Justifica-se a criação de um aplicativo multiplataformas visando atender uma grande diversidade de pessoas, através de seus aparelhos telefônicos, e computadores, facilitando assim a capacidade de controle e monitoramento do diabetes proporcionando funcionalidades rápidas e seguras para se obterem uma melhor qualidade de vida.

Para desenvolver este projeto foi necessário fazer um estudo aprofundado com relação aos cuidados sobre o diabetes, foi utilizado como base o convívio com diversas pessoas portadoras desta doença, onde foi-se analisado os principais pontos de dificuldades ao se conviver com o diabetes. Além do convívio com estas pessoas, foi-se elaborado um estudo bibliográfico para se obter um melhor controle da doença, para que em seguida fosse analisado as principais ferramentas de elicitação de requisitos e recursos para se desenvolver um software eficaz, rápido e seguro. Visando atender uma grande quantidade de pessoas, foi-se desenvolvido um sistema multiplataformas, utilizando o framework Xamarin Forms, que permite a interação com o banco de dados SQLite com a interface, utilizando-se a linguagem de programação C# para seu desenvolvimento.

Portanto pode-se elencar como sendo os principais resultados deste projeto, a construção de um software que necessita de pouco espaço de memória para sua utilização, que tem por finalidade efetuar cálculos complexos e seguros de correções de glicemia e alimentares, exibir relatórios detalhados para se obter um melhor monitoramento da doença, incentivar o usuário a obter um melhor controle através de atividades físicas e o fato de permitir que o usuário possa utiliza-lo sem que seja necessário ter acesso à internet em seu aparelho, pelo fato do sistema utilizar o próprio armazenamento do aparelho.

**2 Levantamento de Requisitos**

Esta seção apresenta a elicitação dos requisitos do projeto, o *business process model and notation* (BPMN), os requisitos funcionais e não funcionais, a matriz de rastreabilidade dos requisitos funcionais e não funcionais, os casos de usos, o diagrama de casos de usos, a matriz de rastreabilidade entre requisitos funcionais e casos de usos e o diagrama da modelagem do banco de dados.

2.1 Elicitação e especificação dos Requisitos

A elicitação dos requisitos do projeto foi feita a partir de técnicas de entrevistas com pessoas portadoras de DM e em reuniões presenciais, onde foram destacadas as principais dificuldades no controle e monitoramento da doença. Também foi utilizado a técnica de etnografia, tendo em vista que o autor deste trabalho possuí a referida doença, atualmente completando 26 anos portando DM. Deste modo, com base na experiência própria do autor e das demais pessoas entrevistadas, foram levantados os principais requisitos para a construção do sistema.

2.2 BPMN

A Figura 2 apresenta o BPMN do projeto, onde pode-se destacar as ações de definir a quantidade de medicamento para corrigir a glicemia, definir a quantidade de medicamento para correções alimentares e exibir relatório no período selecionado.

**Figura 2:** BPMN

Diagrama

Descrição gerada automaticamente

**Fonte:** Autor

2.3 Requisitos Funcionais

O Quadro 1 apresenta os requisitos funcionais do projeto, onde pode-se destacar as ações de cálculo para dosagem de alimentos, cálculo para dosagem de correção de glicemia, sugerir a quantidade de medicamento que deve ser utilizado e a exibição de relatórios.

**Quadro 1 –** Requisitos Funcionais do sistema

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RF001**-Cadastro de dados pessoais | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  (X) Altíssima  ( ) Alta  ( ) Média  ( ) Baixa | | |
| **Descrição**: O sistema deve gravar os dados pessoais de cada usuário sendo estes: Nome, e-mail, celular e o tipo de diabetes que possuí. | | | | |
| **RF002**- Cadastro de medicamentos | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  ( ) Alta  ( ) Média  (X) Baixa | | |
| **Descrição:** O sistema deve registrar o nome dos medicamentos utilizados pelo usuário. | | | | |
| **RF003**- Cadastro de dosagens para correções. | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa | | |
| **Descrição**: O sistema deve registrar a quantidade de insulina referente a determinada quantidade de carboidratos, caso esteja especificado na receita médica.  Exemplo: 1 unidade para cada 15 gramas de carboidrato.  Obrigatoriamente o sistema deve registrar a quantidade de insulina referente ao total do valor de glicemia que deve ser controlado.  Exemplo: 1 unidade para cada 50 mg/dl. | | | | |
| **RF004**- Cadastro do exame de glicemia. | Categoria:  ( ) Oculto  (X)Evidente | | Prioridade:  (X) Altíssima  ( ) Alta  ( ) Média  ( ) Baixa | |
| **Descrição**: O sistema deve registrar dia, horário e o valor informado pelo usuário referente ao exame de glicemia, este valor deverá estar contido entre 21 mg/dl e 599 mg/dl, caso o valor seja inferior a 21 mg/dl o sistema apresentará as siglas LO, e caso o valor seja superior a 599 mg/dl o sistema apresentará as siglas HI. | | | | |
| **RF005-** Calcular média de glicemia. | Categoria:  (X) Oculto  ( ) Evidente | | | Prioridade:  ( ) Altíssima  ( ) Alta  ( ) Média  (X) Baixa |
| **Descrição**: Caso seja solicitado o sistema deve fazer a média da glicemia em determinados períodos definidos pelo usuário.  Fórmula para média glicêmica:  Para cada exame registrado será somado o valor “1” em uma variável “X”.  Em uma segunda variável “Y” será somado o valor obtido através dos registros dos exames de glicemias do usuário.   Deste modo se o usuário registrar de seus exames em 3 horários diferentes, com os seguintes valore: 250; 150; 80 o sistema deverá somá-los e dividi-los pôr 3 chegando ao resultado de 160.   Média = Y/X. | | | | |
| **RF006**- Cálculo de correção de glicemia. | Categoria:  (X) Oculto  ( ) Evidente | | | Prioridade:  (X) Altíssima  ( ) Alta  ( ) Média  ( ) Baixa |
| **Descrição**: Com base nos dados obtidos através do cadastro do usuário, o sistema deve calcular a quantidade de medicamento a ser utilizado.  Desta forma se for registrado um valor hipotético de 200 mg/dl, o sistema deve fazer um cálculo objetivando chegar a um valor entre 70 à 150 mg/dl. | | | | |
| **RF007-** Cadastro de Alimento. | Categoria:  ( ) Oculto  (X) Evidente | | | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa |
| **Descrição:** O usuário do sistema deverá inserir alimentos informando a quantidade de carboidratos e unidade de medida. | | | | |
| **RF008-** Cálculo para dosagem de alimentos. | Categoria:  (X) Oculto  ( ) Evidente | | | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa |
| **Descrição:** O usuário deve informar quais alimentos foram consumidos e suas respectivas quantidades.  Hipoteticamente se o usuário registra um pão francês com equivalência a 15 gramas de carboidratos o sistema deverá definir 1 unidade de insulina para cada pão francês consumido. Este valor irá variar conforme o cadastro feito pelo usuário inicialmente ou conforme for atualizado os dados no sistema. | | | | |
| **RF009-** Sugerir a quantidade de medicamento a ser utilizado | Categoria:  (X) Oculto  ( ) Evidente | | | Prioridade:  (X) Altíssima  ( ) Alta  ( ) Média  ( ) Baixa |
| **Descrição:** Ao se obter os cálculos de correção de glicemia e correção alimentar, o sistema deverá somar os dois resultados e exibir para o usuário a quantidade recomendada de insulina, a sugestão não pode ser um número com casas decimais, deve ser um número inteiro.  Caso o resultado do exame de glicemia for um valor inferior a 70 mg/dl o sistema deve abortar a sugestão, exibindo uma mensagem relatando que o sistema só faz cálculos de sugestões para valores iguais ou superiores à 70 mg/dl. | | | | |
| **RF010-** Informar atividades físicas | Categoria:  ( ) Oculto  (X) Evidente | | | Prioridade:  ( ) Altíssima  ( ) Alta  ( ) Média  (X) Baixa |
| **Descrição:** O usuário pode informar sobre suas atividades físicas diária, onde o sistema deve registrar o dia, horário, nome da atividade realizada e observações feitas pelo usuário. | | | | |
| **RF011-** Monitorar a média de medicamentos consumidos. | Categoria:  (X) Oculto  ( ) Evidente | | | Prioridade:  ( ) Altíssima  ( ) Alta  (X) Média  ( ) Baixa |
| **Descrição:** O sistema deve assim que solicitado pelo usuário fazer o cálculo da média da quantidade de insulina aplicada por um determinado período.  O usuário informa o valor que foi aplicado, o sistema armazena o valor em uma variável “X” e soma o valor 1 em uma segunda variável “Y” para cada registro.  Média = X / Y. | | | | |
| **RF012-** Relatório | Categoria:  (X) Oculto  ( ) Evidente | | | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa |
| **Descrição:** o usuário pode verificar todas as sugestões feitas pelo aplicativo através de um relatório.  O relatório deve conter a média geral de glicemia, quantidade total de exames registrados, total de dosagens aplicadas, média do total de dosagem aplicada, maior e menor índice de glicemia registrado no período selecionado. O sistema também deve gerar os mesmos índices citados anteriormente subdivididos em quatro períodos de 6 horas. | | | | |

**Fonte:** Autor

2.4 Requisitos Não Funcionais

Os requisitos não funcionais do sistema desenvolvido podem ser conhecidos no Quadro 2.

**Quadro 2** – Requisitos Não Funcionais do sistema

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RNF001**-Login | O usuário deverá estar logado para utilizar o sistema. | Tipo Segurança | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF002**- Acessibilidade | Deve ser compatível para Android, IOS e Windows 10. | Tipo Compatibilidade | ( ) Desejável  (X) Obrigatório | ( ) Permanente  (X) Transitório |
| **RNF003-** Facilidade de uso. | O sistema deverá ser de fácil manuseio, tendo uma taxa de até 3 erros por dia no início de sua utilização. | Tipo Usabilidade | (X) Desejável  ( ) Obrigatório | ( ) Permanente  (X) Transitório |
| **RNF004-** Tempo estimado para acesso | O software deve conectar imediatamente o usuário, será estimado uma média de até 30 segundos de espera. | Tipo Desempenho | (X) Desejável  () Obrigatório | ( ) Permanente  (X) Transitório |
| **RNF005-** Tempo estimado para realizar cálculos | O software deverá apresentar os resultados dos cálculos em um tempo estimado de até 15 segundos. | Tipo Desempenho | (X) Desejável  ( ) Obrigatório | ( ) Permanente  (X) Transitório |
| **RNF006-** Disponibilidade de uso | O sistema deverá estar disponível durante as 24h do dia, não sendo necessário estar conectado a uma rede de internet. | Tipo Disponibilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF007-** Identificação do usuário | O usuário deverá identificar qual tipo de diabetes que o usuário possui, se é tipo 1 ou tipo 2 | Tipo Usabilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF008-** Entrega de relatório | O Relatório deve ser entregue assim que solicitado, sendo estimado um tempo de até 30 segundos de espera. | Tipo Usabilidade | ( ) Desejável  (X) Obrigatório | ( ) Permanente  (X) Transitório |
| **RNF09-** Segurança | O usuário apenas pode ver seus próprios dados, não sendo permitido ver dados de terceiros no aplicativo. | Tipo Segurança | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF010-** Armazenamento de dados | O sistema deve comunicar com o banco de dados SQLite. | Tipo Compatibilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF011-** Linguagem de programação | O sistema deve ser desenvolvido utilizando a linguagem C#. | Tipo Compatibilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |

**Fonte:** Autor

O Quadro 3 apresenta a matriz de rastreabilidade entre os requisitos funcionais e requisitos não funcionais.

**Quadro 3** – Matriz de rastreabilidade.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | RF1 | RF2 | RF3 | RF4 | RF5 | RF6 | RF7 | RF8 | RF9 | RF10 | RF11 | RF12 |
| RNF1 |  | X | X | X | X | X | X | X | X | X | X | X |
| RNF2 | X | X | X | X | X | X | X | X | X | X | X | X |
| RNF3 | X | X | X | X | X | X | X | X | X | X | X | X |
| RNF4 | X | X | X | X |  |  | X |  |  | X | X |  |
| RNF5 |  |  |  |  | X | X |  | X |  |  |  |  |
| RNF6 | X | X | X | X | X | X | X | X | X | X | X | X |
| RNF7 | X |  |  |  |  |  |  |  |  |  |  |  |
| RNF8 |  |  |  |  |  |  |  |  |  |  |  | X |
| RNF9 |  | X | X | X | X | X | X | X | X | X | X | X |
| RNF10 | X | X | X | X | X | X | X | X | X | X | X | X |
| RNF11 | X | X | X | X | X | X | X | X | X | X | X | X |

**Fonte:** Autor

2.5 Casos de Uso

A Figura 3 mostra o diagrama de casos de uso utilizado para a criação do projeto.

Índice de casos de uso e Diagrama de casos de uso

UC001 – Gerenciar dados.

UC002 – Informar o resultado do exame de glicemia.

UC003 – Gerenciar alimentos e dosagens.

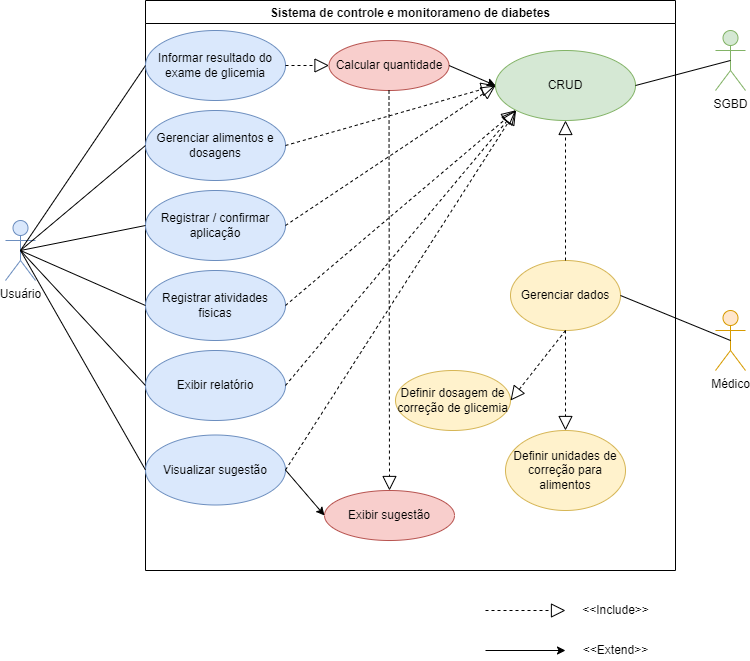
UC004 – Exibir sugestões.

UC005 – Registrar/Confirmar aplicação.

UC006 – Registrar atividades físicas.

UC007 – Exibir relatório.

**Figura 3:** Diagrama de Casos de Uso



**Fonte:** Autor

Os Quadro 5 até o Quadro 11 apresentam mais detalhes sobre os casos de uso do sistema construído.

**Quadro 5** – Use Case Cadastrar Usuários

|  |  |
| --- | --- |
| **Caso de Uso –** Gerenciar dados. | |
| **ID** | UC 001 |
| **Descrição** | Este caso de uso tem como objetivo gerenciar datas e horários para agendamento de consultas e exames que devem ser feitos em laboratórios especializados, e a definição das dosagens de cada medicação definidas pelo profissional de saúde responsável. |
| **Ator Primário** | Paciente |
| **Pré-condição** | É obrigatório que os dados sejam preenchidos com base na receita médica mais recente. |
| **Cenário Principal** | 1. O use case inicia após o usuário selecionar a opção de agendamento. 2. O sistema carrega um formulário de cadastro de horários. 3. O usuário escolhe a opção de agendar um evento ou atualizar evento. 4. O paciente defini os horários e dias quando necessário. 5. O sistema valida as definições. 6. O Sistema armazena os dados. 7. O usuário seleciona a opção configurações. 8. O usuário atualiza as dosagens com base na sua última receita médica. 9. O sistema válido os dados. 10. O sistema encerra o caso de uso. |
| **Pós-condição** | Nenhuma |
| **Cenário Alternativo** | 4a – Horário não informado.  4a.1 O sistema informa ao usuário que este campo é obrigatório e deve ser preenchido.  4a.2 Retorna ao passo 3 do cenário principal.  4b – Horário definido não existe.  4b. 1 O Sistema informa ao usuário que o horário a ser definido não existe.  4b. 2 Retorna ao passo 3 do cenário principal.  4c – Evento sem dia marcado.  4c. 1 O sistema informa ao usuário que para se cadastrar um evento deve-se definir um dia e um horário.  4c. 2 Retorna ao passo 3 do cenário principal.  8a – O usuário não atualiza os dados.  8a. 1 Retorna ao passo 10 do cenário principal. |

**Fonte:** Autor

**Quadro 6** – Informar resultado do exame de glicemia

|  |  |
| --- | --- |
| **Caso de Uso –** Informar resultado do exame de glicemia | |
| **ID** | UC 002 |
| **Descrição** | Este caso de uso tem por objetivo registrar o resultado obtido através do exame de glicemia. |
| **Ator Primário** | Paciente |
| **Pré-condição** | Deverá ser cadastrado no dia e no horário definidos pelo usuário. |
| **Cenário Principal** | 1. O use case inicia ao selecionar a opção cadastro de exames. 2. O sistema carrega o formulário de cadastro. 3. O usuário informa o dia, o horário e o resultado do exame de glicemia. 4. O sistema valida os campos. 5. O sistema armazena os dados e informa ao usuário que os dados foram salvos. 6. O sistema encerra o caso de uso. |
| **Pós-condição** | Nenhuma |
| **Cenário Alternativo** | 4a – Dia e horário não preenchidos ou inválidos  4a.1 O sistema informa que os campos dia e horário são obrigatórios e que devem ser preenchidos com valores válidos.  4a. 2 Retorna ao passo 3 no cenário principal.  4b – Exame de glicemia inválido ou não preenchido.  4b. 1 O sistema informa que o exame de glicemia deve ser um valor válido sendo estes valores LO, HI ou algum valor numérico entre 21mg/dl à 599 mg/dl, e notifica que este campo é obrigatório.  4b. 2 Retorna ao passo 3 no cenário principal. |

**Fonte:** Autor

**Quadro 7** – Gerenciar alimentos e dosagens

|  |  |
| --- | --- |
| **Caso de Uso –** Gerenciar alimentos e dosagens**.** | |
| **ID** | UC 003 |
| **Descrição** | Este caso de uso tem por objetivo gerenciar a quantidade de insulina a ser aplicado para controlar os valores de carboidratos dos alimentos consumidos. |
| **Ator Primário** | Paciente |
| **Pré-condição** | O alimento em análise deve estar cadastrado no sistema. |
| **Cenário Principal** | 1. O use case inicia após o usuário selecionar a opção de contagem de carboidratos. 2. O sistema carrega o formulário de seleção de alimentos. 3. O usuário seleciona/cadastra um alimento. 4. O sistema abre um formulário para inserir um novo alimento. 5. O usuário preenche os campos destacando quais alimentos foram consumidos e sua respectiva quantidade. 6. O sistema valida os campos. 7. O sistema calcula a quantidade total de carboidratos informadas armazena os dados obtidos. 8. O sistema verifica se o paciente faz uso de insulina injetável, caso faça o valor obtido dos carboidratos é convertido para unidades do medicamento. 9. O sistema informa ao usuário que foi concluído essa etapa com sucesso. 10. O sistema encerra o caso de uso. |
| **Pós-condição** | Nenhuma |
| **Cenário Alternativo** | 6a – Alimento não cadastrado.  6a.1 O sistema informa ao usuário que o alimento informado não está cadastrado no banco de dados.  6a.2 Retorna para o passo 3 do cenário atual. |

**Fonte:** Autor

**Quadro 8** – Exibir sugestões

|  |  |
| --- | --- |
| **Caso de Uso – Exibir sugestões.** | |
| **ID** | UC 004 |
| **Descrição** | Este caso de uso tem por objetivo sugerir a quantidade correta de da medicação a ser utilizada. |
| **Ator Primário** | Paciente |
| **Pré-condição** | Os dados serão apresentados somente após ser executado pelo menos o UC002. |
| **Cenário Principal** | 1. O use case inicia após o usuário informar o resultado do exame de destro e/ou os alimentos consumidos. 2. O sistema valida os campos obrigatórios 3. O sistema deverá calcular a quantidade de insulina/comprimido, tendo como referência o resultado do destro e a quantidade de carboidratos ingeridos. 4. O sistema sugere a quantidade de medicamento. |
| **Pós-condição** | Nenhuma. |
| **Cenário Alternativo** | 2a – Resultado do destro não informados  2a. 1 O sistema informa que o atributo resultado de destro é obrigatório.  2b. 2 Retorna ao passo 1 no cenário principal |

**Fonte:** Autor

**Quadro 9** – Registrar/Confirmar aplicação

|  |  |
| --- | --- |
| **Caso de Uso –** Registrar/Confirmar aplicação | |
| **ID** | UC 005 |
| **Descrição** | Este caso de uso tem por finalidade fazer o cadastro das quantidades das medicações feitas pelo usuário |
| **Ator Primário** | Paciente |
| **Pré-condição** | Deve ter sido apresentado a sugestão ao usuário. |
| **Cenário Principal** | 1. O use case inicia após o sistema exibir a sugestão ao usuário.   1. O sistema abre um formulário de observações. 2. O usuário, anota a quantidade que foi aplicada caso seja diferente da sugestão. 3. O sistema carrega uma caixa de confirmação. 4. O usuário confirma a aplicação. 5. O sistema salva os dados. 6. O sistema encerra o caso de uso. |
| **Pós-condição** | Nenhuma |
| **Cenário Alternativo** | 4a – Dados não confirmados.  4a.1 O sistema encerra o caso de uso e retorna uma tela de notificação. |

**Fonte:** Autor

**Quadro 10** – Registrar atividades físicas

|  |  |
| --- | --- |
| **Caso de Uso –** Registrar atividades físicas. | |
| **ID** | UC 006 |
| **Descrição** | Este caso de uso tem por objetivo monitorar as atividades físicas realizadas pelo usuário. |
| **Ator Primário** | Paciente |
| **Pré-condição** | Nenhuma. |
| **Cenário Principal** | 1. O use case inicia após o sistema alertar o usuário de suas atividades físicas. 2. O sistema carrega o formulário de cadastro. 3. O usuário informa qual atividade física foi realizada (caminhada, corrida, futebol, luta, etc.) e também informa o período que foi realizada a atividade. 4. O sistema valida os campos obrigatórios. 5. O sistema salva os dados no banco de dados e informa ao usuário que os dados foram salvos. 6. O sistema encerra o caso de uso. |
| **Pós-condição** | Nenhuma |
| **Cenário Alternativo** | 1a – Usuário ignora o alerta.  1a – O sistema encerra o caso de uso.  4a – Campos Atividade realizada não preenchido.  4a.1 O sistema informa ao usuário que este campo é obrigatório e deve ser preenchido.  4a.2 Retorna ao passo 3 do cenário principal. |

**Fonte:** Autor

**Quadro 11** – Exibir relatório

|  |  |
| --- | --- |
| **Caso de Uso –** Exibir relatório | |
| **ID** | UC 007 |
| **Descrição** | Este caso de uso tem por objetivo permitir que o usuário possa monitorar a doença. |
| **Ator Primário** | Paciente |
| **Pré-condição** | Deverá conter pelo menos um dado cadastrado. |
| **Cenário Principal** | 1. O use case inicia após o usuário selecionar a opção relatório. 2. O sistema valida os dados que deveram conter no relatório. 3. O sistema gera o relatório. 4. O sistema encerra o caso de uso. |
| **Pós-condição** | Nenhuma |
| **Cenário Alternativo** | 2a – Nenhum exame de destro registrado.  2a.1 O sistema informa ao usuário que é necessário ter pelo menos um exame de destro cadastrado para gerar o relatório  2a.2 Retorna ao passo 1 do cenário principal. |

**Fonte:** Autor

**Quadro 12** – Matriz de rastreabilidade RF x UC

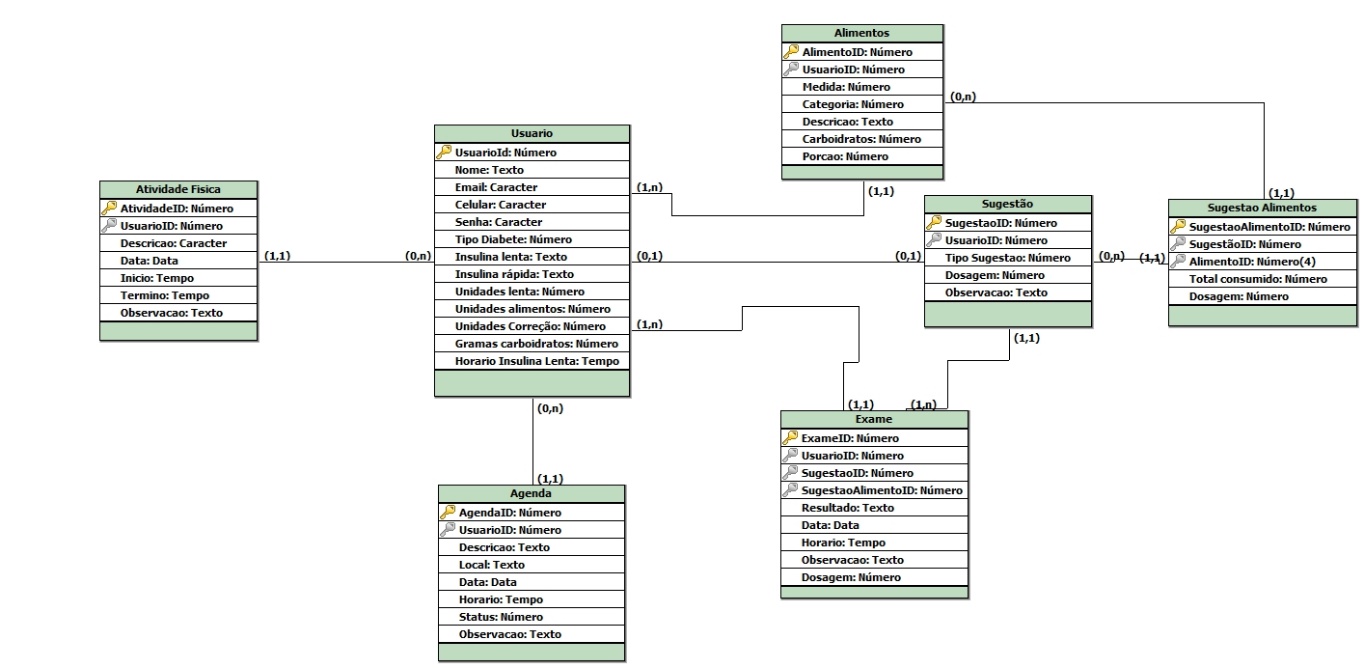
|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | RF 1 | RF 2 | RF 3 | RF 4 | RF 5 | RF 6 | RF 7 | RF 8 | RF 9 | RF 10 | RF 12 |
| UC1 |  | X | X |  | X |  | X | X |  |  | X |
| UC2 | X |  |  |  |  |  |  |  |  |  |  |
| UC3 |  |  | X |  |  | X | X | X |  |  |  |
| UC4 |  |  |  |  |  |  |  | X |  |  |  |
| UC5 |  |  | X |  | X |  | X | X | X |  |  |
| UC6 |  |  |  |  |  |  |  |  | X |  |  |
| UC7 |  |  |  |  |  |  |  |  |  |  | X |

**Fonte:** Autor

2.6 Diagrama de modelagem do banco de dados

A Figura 4 apresenta o diagrama que modela o banco de dados do sistema construído.

**Figura 4:** Diagrama de Modelagem do Banco de Dados



**Fonte:** Autor

**3 Ferramentas e Métodos ou Desenvolvimento**

Este capítulo apresenta as ferramentas utilizadas para a construção do sistema, descreve como foi sua implementação e os métodos utilizados para o desenvolvimento.

3.1 Ferramentas e Implementação

A linguagem de programação escolhida para o desenvolvimento do projeto foi a C#, utilizando-se a IDE Visual Studio Community 2019, por facilitar os *backups* incrementaís feitos no projeto através da plataforma Git-Hub, além de propiciar a instalação completa do framework Xamarin-Forms na versão 5.0.0.2115.

A escolha do *framework* Xamarin-Forms se tornou necessária para distribuir a mesma linha de código para diversas plataformas diferentes, sendo elas o sistema operacional Windows 10, Android e IOS, tendo o principal foco na plataforma Android.

Foi utilizado o banco de dados SqLite na versão 1.8.116, que possibilita a gravação dos dados no próprio dispositivo do usuário, dessa forma não sendo necessário a utilização de Internet para usufruir dos benefícios do sistema, sendo esta a principal razão para o seu uso.

3.1.1 Linguagem de programação C#.

C# é uma linguagem orientada a objetos desenvolvida por Anders Hejlsberg para a Microsoft, fazendo parte da aplicação .NET. Em primeira instância a linguagem foi batizada como Cool no final do século XX, mais tarde após a virada do século a linguagem Cool foi renomeada para C#.

Após o renome da linguagem, a Microsot submeteu-se à ECMA (European Computer Manufacturers Association), associação cujo objetivo é a padronização de sistemas de informação. Em 2001, a ECMA aprovou o C# e a linguagem recebeu a especificação ECMA-334. Mais tarde, em 2003, tornou-se padrão da ISO, recebendo a especificação de ISO/IEC 23270 (COIMBRA, 2013).

O C# surgiu com o propósito de tornar menos rígido o desenvolvimento de aplicativos e foi baseada nas linguagens de programação Java, C e C++. Unindo diversos recursos de cada linguagem citada a fim de se tornar mais simples e flexível e introduzir diversos elementos como: expressões lambda, tipos primitivos com valores nulos, delegações e acessos diretos a memória.

Em relação a linguagem C, o C# tem foco na compilação de soluções de alto nível, já a linguagem C é voltada para o desenvolvimento de baixo nível. Outras comparações, de acordo com MSDN (2012), o C#, simplifica de modo significativo a complexidade do C++ e introduz novos elementos não disponíveis no Java, tais como: tipos primitivos não nulo, delegações, expressões lambda e acesso direto à memória.

MSDN (2012) ressalta que, o processo de compilação do C# é mais simples e mais flexível comparado ao C++ ou Java, pois elimina-se a necessidade de arquivos de cabeçalhos separados e não há a necessidade da declaração de métodos e tipos em uma ordem específica. Um arquivo contendo códigos de programação podem ser definidos quaisquer números de classes, estruturas, interfaces e eventos que forem necessários. (COIMBRA, 2013).

3.1.2 Banco de dados SQLite.

Na prática o SQLite é um banco de dados gratuito que se assemelha à um Sistema de Gerenciamento de Banco de Dados, possibilitando fazer inserções, alterações e exclusões de dados (CRIS, 2015). O SQLite é formado por um conjunto de bibliotecas escritas em C, com a finalidade de ser inserido em diversos programas escritos em diversas linguagens.

A principal diferença com outros bancos de dados, é que a manipulação dos dados pode ser feita sem que seja preciso acessar um SGBD (sistema de gerenciamento de banco de dados), ou seja, todas as instruções podem ser feitas através do código fonte, inclusive a criação do banco de dados e tabelas, sendo esse o principal motivo por ser implementada em aplicativos móveis para Android e IOS.

Além disso, o Android oferece suporte completo ao banco, através de uma API com um rico conjunto de classes e métodos que abstraem as complexidades dos códigos SQL. Assim, não precisamos montar a cláusula SQL inteira para atualizar uma linha na tabela, ou ainda, para fazer uma pesquisa na mesma. O Android nos fornece um método, onde passando alguns parâmetros e obtemos um apontador para os dados retornados, podendo navegar pelo resultado como se estivéssemos escolhendo uma folha em um arquivo (CRIS, 2015).

3.1.3 Xamarin.Forms

O Xamarin é uma plataforma de software livre de desenvolvimento utilizada para criar sistemas em diversas plataformas diferentes, o framework é uma extensão da plataforma de desenvolvimento .NET que é composta por diversas bibliotecas, ferramentas e linguagens de programação (JOHNSON et al., 2022).

O Xamarin.Forms é uma estrutura do Xamarin que tem por finalidade disponibilizar bibliotecas e ferramentas para o desenvolvimento de aplicativos multiplataformas (OGLIARI, 2017). O Xamarin.Forms é uma estrutura de código aberto da Microsoft, tendo seu principal objetivo a criação de aplicativos para Android, IOS e sistemas operacionais Windows através do .NET com uma única base de códigos compartilhados.

3.2 Métodos ou Desenvolvimento

Foi utilizado o padrão MVVM (*Model-View-Viewmodel*) para o desenvolvimento do aplicativo. “O MVVM foi criado em 2005 por John Gossman um dos arquitetos da equipe do WPF e Silverlight na Microsoft.” (DE OLIVEIRA, 2011). Ele se assemelha ao padrão MVC (*Model-View-Controller*), contudo procura estabelecer uma separação mais apurada de responsabilidades em uma aplicação WPF e Silverlight.

Em termos gerais a View é responsável por definir o *front-end* do sistema, que faz uma interligação com a ViewModel através de um componente denominado de DataContext. A ViewModel tem como principal responsabilidade a criação de uma lógica de apresentação para a View, fazendo a interligação entre View e Model. Por sua vez, a Model tem a responsabilidade de encapsular a lógica e os dados.

A Figura 5 mostra um exemplo da construção do *front-end* em uma View para a construção da tela de login. É possível perceber na linha 15 a manipulação de imagem, sendo definido altura e largura. Nota-se que todos os elementos estão contidos dentro de um *frame*, que permite destacar a área contida, causando efeitos 3D nas plataformas Android e IOS.

**Figura 5 –** View

Texto

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 6 mostra um exemplo de uma ViewModel. A linha 18 mostra a definição da interface INotifyPropertyChanged, que é responsável por proporcionar um mecanismo unificado para implementar um único evento para todos os eventos dos objetos utilizados. Como exemplo prático temos as validações que são feitas de forma dinâmica apresentando a mensagem de erro e impossibilitando o envio do formulário antes mesmo do usuário clicar no botão de confirmação.

**Figura 6 –** ViewModel

Texto

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 7 é um exemplo de Model, que é responsável por encapsular todos os dados utilizados na aplicação. Por meio da model definimos qual tabela será utilizada no banco de dados e quais são as entidades envolvidas.

**Figura 7 –** Model

Texto

Descrição gerada automaticamente

**Fonte:** Autor

Para que fosse possível integrar a mesma linha de código para diversas plataformas diferentes foi implementada uma interface com a finalidade de direcionar os dados para cada plataforma específica.

**Figura 8 –** Interface

Texto

Descrição gerada automaticamente

**Fonte:** Autor

A interface se comunica com as classes contidas em cada plataforma e possibilita a reutilização do mesmo código. Além disso, em cada classe é efetuado o código para cada tipo de plataforma retornando um resulta. Na Figura 8 temos a manipulação de dados no banco de dados SQLite.

**Figura 9 –** Classe Da plataforma Android

Texto

Descrição gerada automaticamente

**Fonte:** Autor

**4 Resultados e Discussão**

Neste capítulo serão apresentadas as principais funcionalidades construídas no projeto, com destaque para as funcionalidades de cadastro de alimentos, cadastro de índice de glicemia, obtenção de sugestão de quantidade de medicamento e o relatório detalhado de um período selecionado.

4.1 Tela de termos de responsabilidades

A Figura 10 mostra a notificação dada ao usuário sobre o uso das funcionalidades e a obrigatoriedade de se utilizar a receita médica para preenchimento dos dados cadastrais. A tela explica quais situações não serão feitos os cálculos de correção de glicemia para prevenir casos de hipoglicemias severas por conta do medicamento utilizado de maneira incorreta. Também deixa explicito ao usuário que, ao se obter o valor “HI” na medição de glicemia, o sistema utilizará o valor de 600 mg/dl como base para efetuar os cálculos, isto porque, atualmente, os aparelhos de medição de glicemia conseguem registrar somente até o valor de 599 mg/dl e apresentam valores superiores com a palavra “HI”.

**Figura 10 –** Termos de responsabilidades

Texto

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 11 apresenta a tela de cadastro de medicamentos, onde o usuário definirá qual é o seu tipo de diabetes por meio de uma caixa de seleção com os valores: diabetes tipo 1, diabetes tipo 2 e diabetes gestacional. Na mesma tela o usuário definirá o nome da insulina de efeito lento, juntamente com sua respectiva quantidade e o horário que é utilizado este medicamento, também nessa tela o usuário deverá cadastrar o nome da insulina de efeitos rápidos, que é a responsável por realizar as correções de glicemia.

**Figura 11 –** Tela de cadastro de medicamentos

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 12 mostra a tela de cadastro de medicamentos para correção de glicemia. Aqui o usuário informa quantas unidades de insulina deverão ser aplicadas para cada determinada quantidade de carboidratos e unidades de glicemia, tendo como base a receita médica mais atual.

**Figura 12 –** Tela de cadastro de correções

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 13 mostra a tela de cadastro de alimentos, onde o usuário pode cadastrar os alimentos que geralmente fazem parte de sua alimentação diária e para que seja possível obter o cálculo de correção alimentar. Nessa tela é preciso cadastrar o nome do alimento, a unidade de medida (unidades, gramas, ml), a porção indicada nos rótulos das embalagens dos alimentos, as respectivas quantidades de carboidratos existentes para cada porção e em qual categoria alimentícia o alimento se enquadra.

**Figura 13 –** Tela de cadastro de alimentos

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 14 apresenta a tela de busca de alimentos cadastrados. A partir desta tela o usuário pode abrir o formulário para cadastrar um novo alimento e realizar a busca pelo alimento desejado usado como parâmetros o nome do alimento, a categoria a que pertence ou unidades de medidas. Na listagem dos alimentos é definido o nome do alimento, a quantidade de carboidratos que possui, a categoria a que pertence e sua unidade de medida. Ainda é possível realizar exclusões ou edições nos dados por meio dos respectivos botões.

**Figura 14 –** Tela de consulta de alimentos

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 15 apresenta a tela de cadastros de exames de glicemia, que tem por finalidade registrar os dados obtidos em cada medição de glicemia. O usuário informa o dia e horário que foi realizado o exame, qual foi o resultado do exame (valores numéricos entre 20 e 599 mg/dl) e, caso o valor obtido pelo aparelho de medição de glicemia apresentar as siglas “LO” ou “HI”, o usuário deverá clicar no botão referente a esses resultados. Nesta tela também é possível informar a quantidade de insulina aplicada, sendo este um dado opcional a ser preenchido, e também conta com um campo de observações.

**Figura 15 –** Tela de cadastro simples de resultado de exame de glicemia para monitoramento

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 16 apresenta a tela onde são inseridos os dados para obtenção da sugestão de quantidade de medicamento a ser utilizada. O usuário define o tipo de cálculo que o sistema deve realizar (correção de glicemia, correção de alimentos ou correção de glicemia e alimentos). Caso seja selecionada a opção “correção de alimentos” ou “correção de glicemia e alimentos” o sistema exibirá uma busca onde devem ser informados os alimentos consumidos e suas respectivas quantidades. Na tela ainda há um campo para observações.

**Figura 16 –** Tela de cadastro de exames de glicemia e sugestões de medicamentos

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 17 mostra uma simulação de preenchimento de dados de resultado de glicemia, a funcionalidade de busca de alimentos no banco de dados SQLite com quantidades consumidas e uma sugestão de medicamentos a ser utilizada a partir desses dados. Após realizar os cálculos o sistema apresenta na mesma tela a sugestão da quantidade de medicamentos e permite que o usuário informe a quantidade de medicamentos realmente utilizada, caso o valor seja diferente da sugestão.

**Figura 17 –** Simulação de cadastro de exame de glicemia e obtenção da sugestão de medicamentos

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 18 apresenta a tela de registros dos exames e sugestões cadastradas pelo usuário. Nesta tela encontram-se os botões que direcionaram o usuário para as telas de registro de exame e registro de exames com a obtenção da sugestão de medicamentos. É permitido filtrar os registros por datas, resultado de exames de glicemia ou valor da sugestão de dosagem realizada pelo sistema.

A *grid* lista todos os exames registrados em ordem decrescente e apresenta a data e horário que foi cadastrado, o resultado do exame, o valor da sugestão de dosagem feita pelo sistema e quantidade de medicamento que o usuário consumiu.

**Figura 18 –** Tela de consulta de registros de exames de glicemia e sugestões

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 19 apresenta a tela de cadastro de atividades físicas realizadas pelo usuário, onde é possível gravar os dados referentes à data de realização da atividade física, o nome da atividade, o horário de início e término e registro de observações.

**Figura 19 –** Tela de cadastro de atividades físicas

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 20 mostra a tela de listagem dos registros de atividades físicas realizadas. A tela possui um botão que direciona para o cadastro de atividades físicas e uma funcionalidade para filtrar as atividades físicas por nome ou data. A *grid* que exibe os dados apresenta a data de registro, nome da atividade, hora de início, hora de término e observações. Também é possível acessar a funcionalidade de exclusão e alteração de registros de atividades.

**Figura 20 –** Tela de listagem de registros de atividades físicas

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 21 apresenta a tela de agendamento de consultas e exames a serem feitos em laboratório. O usuário consegue cadastrar a descrição do agendamento, o local onde se encontra o consultório e/ou laboratório, a data, hora e observações a respeito do evento. Na mesma tela pode-se verificar os registros de agendamentos, com possibilidade de filtrar por descrição ou data do evento. Na listagem pode-se verificar o nome do evento, o dia e horário e, na mesma *grid,* é permitido alterar o *status* do evento de “pendente” para “concluído”. Também é possível excluir os registrar dos eventos na mesma *grid*.

**Figura 21 –** Tela de agendamento de consultas e exames de laboratório

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Fonte:** Autor

A Figura 22 mostra a tela de relatórios de exames cadastrados no sistema. Nesta funcionalidade o usuário consegue gerar um relatório detalhado dos dados cadastrados no sistema e aplicar filtros por datas. Após selecionar o período desejado o sistema informa ao usuário a média geral de glicemia e o total de registros feitos nas funcionalidades de sugestão de medicamento e cadastro de exames de glicemia. Informa também as unidades da insulina de ação lenta cadastradas, a dosagem média de insulina de ação rápida utilizada no período de 24 horas e exibe o maior e menor valores de resultado de exame de glicemia registrado no período filtrado, com dia e horário do registro.

Na mesma funcionalidade o sistema gera informações detalhadas com os mesmos dados citados anteriormente, com exceção das unidades de insulina de ação lenta, porém, referindo-se a quatro períodos pré-determinados pelo sistema: das 00h00 às 06h00, das 06h01 às 12h00, das 12h01 às 18h00 e das 18h01 às 23h59. Desta forma o sistema consegue monitorar melhor a doença no dia a dia do usuário, identificando períodos de possíveis alterações de dosagens de medicamentos.

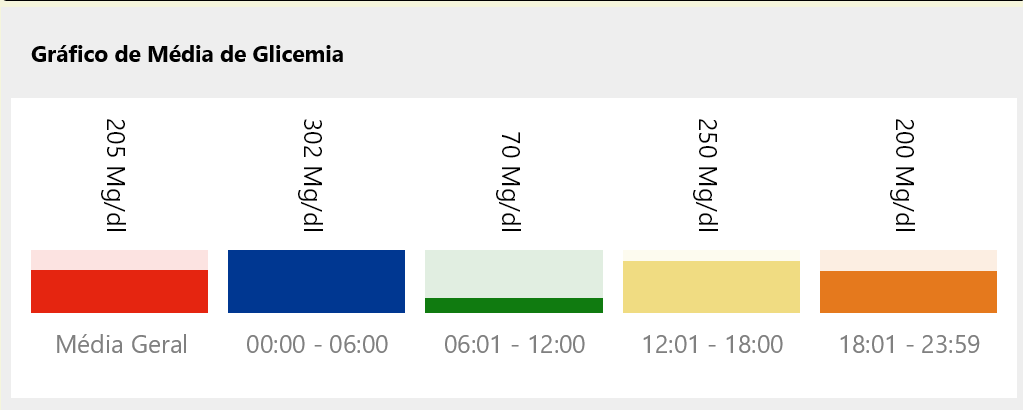
**Figura 22 –** Tela de relatórios



**Fonte:** Autor

A Figura 23 mostra o gráfico de média de glicemia gerado ao executar a funcionalidade de relatórios. Os gráficos mostram valores referentes à média geral de glicemia, média do período das 00h00 às 06h00, média do período das 06h01 às 12h00, média do período das 12h01 às 18h00 e média do período das 18h01 às 23h59.

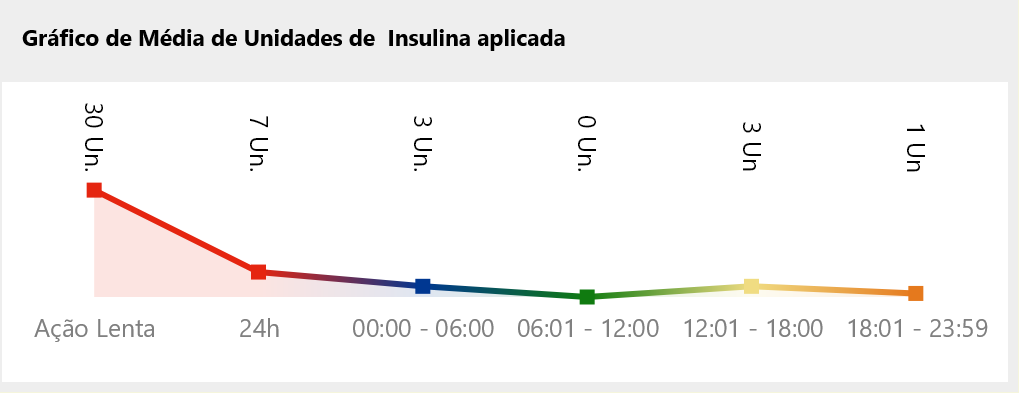
**Figura 23 –** Gráficos de média de glicemia na tela de relatórios



**Fonte:** Autor

A Figura 24 mostra o gráfico de média de unidades de insulina aplicadas no período selecionado. O primeiro valor refere-se à quantidade de insulina de ação lenta registrada no sistema, o segundo valor refere-se à média de insulina aplicada no período de 24 horas, os demais valores referem-se às quantidades insulina aplicas nos períodos determinados pelo sistema (00h00 às 06h00, 06h01 às 12h00, 12h01 às 18h00 e 18h01 às 23h59).

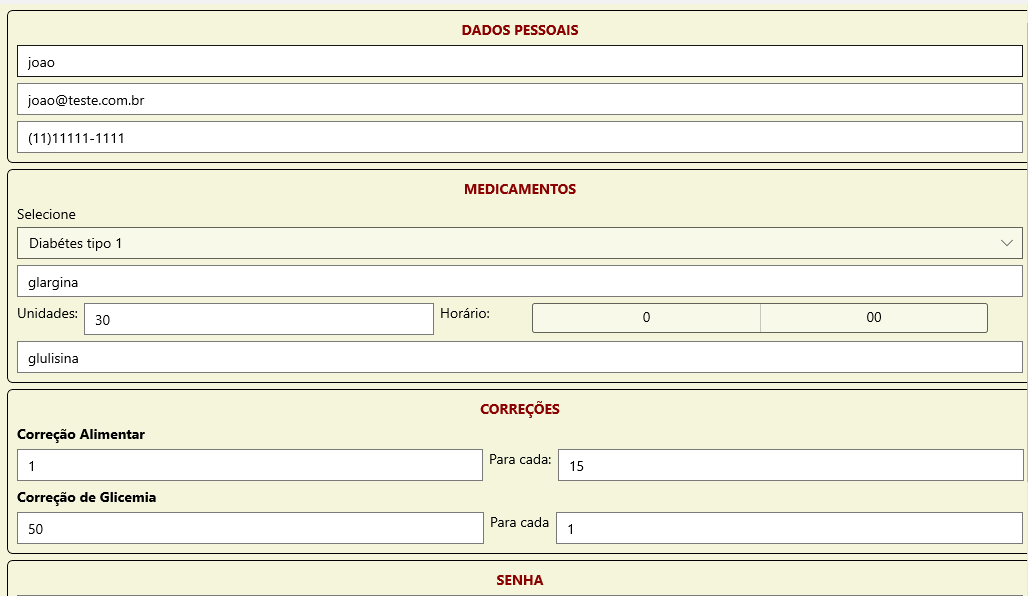
**Figura 24 –** Gráficos de média de unidades de insulina aplicada na tela de relatórios



**Fonte:** Autor

A Figura 25 apresenta a tela de configurações do sistema. Nesta funcionalidade o usuário pode alterar seus dados pessoais (nome, e-mail e celular) e atualizar os dados de seus medicamentos. Na área de edição de dados sobre medicamentos o usuário pode alterar o tipo de diabetes que possui por meio de uma caixa de seleção com os valores diabete tipo 1 e diabetes tipo 2, pode alterar o nome da insulina de ação lenta, alterar as unidades e horário de aplicação da insulina de ação lenta e definir um novo nome para insulina de ação rápida. Nessa mesma tela o usuário pode atualizar os dados de correções de glicemia e de alimentação e também alterar sua senha.

**Figura 25 –** Tela de configurações



**Fonte:** Autor

O Capítulo 4 apresentou as principais funcionalidades implementadas para se obter um melhor controle e monitoramento do diabetes mellitus.

**Considerações finais**

O mau controle e monitoramento da doença diabetes mellitus acarreta diversos problemas de saúde a quem possuí tal doença, sendo que em alguns casos pode-se levar a pessoa a óbito. Sendo assim o objetivo deste projeto foi apresentar o desenvolvimento de um aplicativo para controle das taxas de glicemia no intuito de proporcionar uma melhor qualidade de vida a pessoas portadoras da comorbidade diabete mellitus. O sistema auxilia o usuário a corrigir seu índice de glicemia com base em receita médica realizando cálculos complexos de correções de glicemia e alimentares, em seguida informando a sugestão da quantidade de medicamentos que deverá ser consumido.

Inicialmente se propôs a criação de um sistema web que deveria fazer registros do índice de glicemia, horário e observações, e exibir uma sugestão de medicamento para que se fosse possível realizar o controle da doença. Porém, a partir da convivência com a doença, e o convívio com outras pessoas que portam esta mesma doença, e tendo o conhecimento que não existe cura, foi-se levantado os principais requisitos para proporcionar facilidades e um melhor monitoramento no controle diário desta comorbidade. Na primeira fase do projeto foi identificado quais as principais formas de controle, e como são feitos estes controles pelos pacientes, em seguida foi estudado o que mais é levado em consideração a respeito de controlar esta doença.

Após o levantamento dessas pesquisas, começou um estudo para o desenvolvimento multiplataformas de um sistema que pode ser utilizado no próprio computador, IOS e android dessa forma ampliando ainda mais a possibilidade de uma pessoa portadora de diabetes a vir utilizar este software, sendo assim foi-se utilizado o framework Xamarin Forms que possibilitou desenvolver este projeto utilizando a linguagem de programação C#.

Para que se fosse possível desenvolver este software, foi-se necessário um estudo aprofundado sobre a linguagem de programação C#, bem como o estudo sobre o banco de dados SQLite, para que fosse desenvolvido um software com maior qualidade e profissionalismo. Sendo assim foi-se necessárias diversas horas de estudos e certificações, para conseguir concluir o desafio de lidar com uma linguagem e banco de dados que não são utilizadas nas grades do curso.

Visando melhorar o projeto, futuramente pode-se implementar as funcionalidades de envio de e-mail ao cadastrar e recuperar senha, implementação de uma funcionalidade de envio de alertas ao agendar uma atividade no software, podendo implementar esta mesma funcionalidade caso o usuário defina os horários que deveram ser feitos os exames de glicemia, implementação de arquivamento de dados contidos no aparelho do usuário em um banco de dados externo, com a finalidade de garantir uma melhor segurança em relação a back-ups, e a implementação de uma funcionalidade que permita a interação do software diretamente com o glicosímetro, de tal maneira que o usuário não precise digitar o valor obtido na tela do sistema.

**Referências**

BEZERRA, Clarisse. **Cetoacidose diabética**: o que é, sintomas e tratamento. [*S. l.*], outuro 2021. Disponível em: https://www.tuasaude.com/cetoacidose-diabetica/. Acesso em: 1 maio 2022.

BEZERRA, Clarisse. **Glicemia capilar**: o que é, como medir e valores de referência. [*S. l.*], junho 2021. Disponível em: https://www.tuasaude.com/como-medir-a-glicemia/#:~:text=Como%20medir%20a%20glicemia%20capilar&text=Inserir%20uma%20fita%20de%20teste,apare%C3%A7a%20no%20monitor%20do%20aparelho. Acesso em: 30 abr. 2022.

BRASIL. Ministério da Saúde. **Diabetes Mellitus.** Brasília, 2006.

BRASIL. Ministério da Saúde. **RASTREAMENTO E DIAGNÓSTICO DE DIABETES MELLITUS GESTACIONAL NO BRASIL**. Brasília, 2017.

BRASIL. Ministério da Saúde. **Relatório de Recomendações.** Brasília, 2009.

COIMBRA DE ARAÚJO, Everson. **Artigo Invista em você! Saiba como a DevMedia pode ajudar sua carreira. A evolução da linguagem de programação C#**. [*S. l.*], 2013. Disponível em: https://www.devmedia.com.br/a-evolucao-da-linguagem-de-programacao-csharp/28639. Acesso em: 15 out. 2021.

CRIS BRITO, Robson. **Utilizando SQLite em aplicativos Android**. [*S. l.*], 2015. Disponível em: https://www.devmedia.com.br/utilizando-sqlite-em-aplicativos-android/32117. Acesso em: 15 out. 2021.

DALTOÉ, Luciane. **Diabetes e Alimentação**: Com a Nutricionista profª Dra. Luciane Daltoé. [*S. l.*], 26 jun. 2013. Disponível em: http://www.unesc.net/portal/blog/ver/73/23336. Acesso em: 30 abr. 2022.

DE OLIVEIRA, Fabio. **Introdução ao MVVM (Model-View-ViewModel)**. [*S. l.*], 18 abr. 2011. Disponível em: https://fabiosoliveira.wordpress.com/2011/04/18/introducao-ao-mvvm-model-view-viewmodel/. Acesso em: 30 abr. 2022.

Disponível em: https://imasters.com.br/desenvolvimento/plataformas-para-desenvolvimento-de-apps-moveis-parte-03-desenvolvimento-mobile-multiplataforma-com-xamarin. Acesso em: 30 abr. 2022.

JOHNSON, Justin; ARYA, Hemant; BRITCH, David; DUNN, Craig. **O que é o Xamarin?**. [*S. l.*], 2 abr. 2022. Disponível em: https://docs.microsoft.com/pt-br/xamarin/get-started/what-is-xamarin. Acesso em: 30 abr. 2022.

MINISTÉRIO DA SAÚDE (Brasil). Biblioteca Virtual em Saúde. **Diabetes**. [*S. l.*], junho 2009. Disponível em: https://bvsms.saude.gov.br/diabetes/. Acesso em: 1 maio 2022.

**Manual oficial de contagem de carboidratos regional** / Sociedade Brasileira de Diabetes, Departamento de Nutrição. – Rio de Janeiro: Dois C: Sociedade Brasileira de Diabetes, 2009 il.

OGLIARI, Ricardo. **Plataformas para desenvolvimento de apps móveis – Parte 03**: Desenvolvimento Mobile Multiplataforma com Xamarin. [*S. l.*], 20 dez. 2017.

1. Graduando em Análise e desenvolvimento de sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: joaopaulo.fr.1994@gmail.com [↑](#footnote-ref-1)