

SISTEMA DE DESENVOLVIMENTO DE HÁBITOS

BetterYou

Nome do projeto	BetterYou
Nome do time	NPC's
Integrantes	Emily Perin, Gabrielle Tomaszewski, Rodrigo Ruoso Puntel, Thiago Rocha Frantz

Versionamento do documento

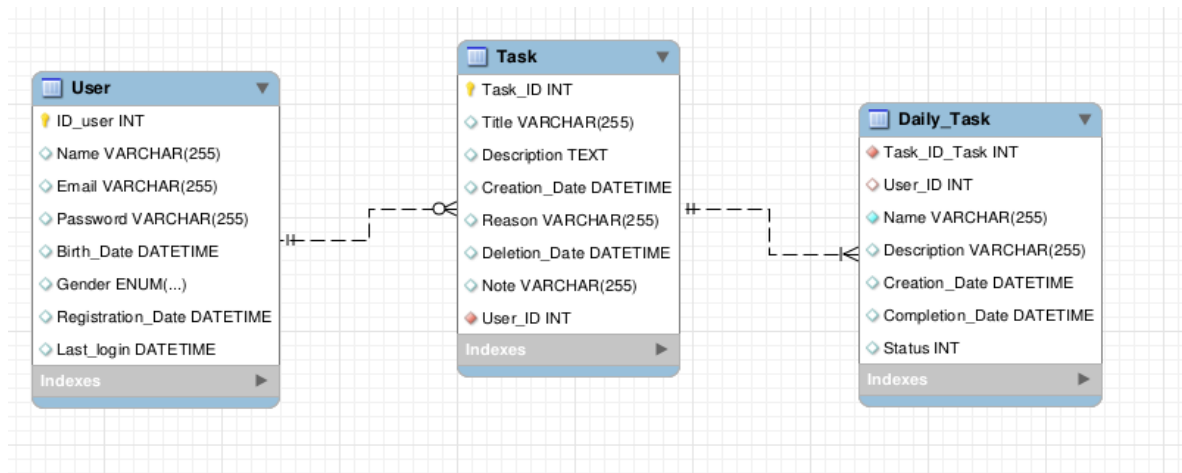
Versão	Data	Responsável	Observações
...			
1.3			
1.2			...
1.1	15/06/2024		Entrega parcial fornecida pelo time NPC's
1.0	24/05/2024	Professores Evandro Franzen e Kurt Molz	Primeira versão publicada do documento com visão geral do sistema.

SUMÁRIO

1. MODELO ER	3
1.1 Imagem do Modelo	3
2. SCRIPTS	4
3. ESPECIFICAÇÃO DAS INTERFACES	4
3.1 Escolher uma transação	4
3.2 Escolher uma transação	4
4. ESPECIFICAÇÃO DAS REGRAS DE NEGÓCIO	4
4.1 Usar a mesma transação da interface	5
4.2 Usar a mesma transação da interface	5
5. INCREMENTOS DOS ANALISTAS	5

1. MODELO ER

1.1 Imagem do Modelo



2. SCRIPTS

2.1 Criação do BD

Incluir aqui o script que gera todo o banco de dados do sistema.

```
CREATE DATABASE BetterYouHabits;
```

```
USE BetterYouHabits;
```

```
CREATE TABLE User (
    User_ID INT PRIMARY KEY AUTO_INCREMENT,
    Name VARCHAR(255) NOT NULL,
    Email VARCHAR(255) NOT NULL UNIQUE,
```

```
    Password VARCHAR(255) NOT NULL,  
    Birth_Date DATETIME NOT NULL,  
    Gender ENUM('Male', 'Female', 'Non-Binary', 'Other'),  
    Registration_Date DATETIME DEFAULT CURRENT_TIMESTAMP,  
    Last_Login DATETIME  
);
```

```
CREATE TABLE Task (  
    Task_ID INT PRIMARY KEY AUTO_INCREMENT,  
    Title VARCHAR(255) NOT NULL,  
    Description TEXT,  
    Creation_Date DATETIME DEFAULT CURRENT_TIMESTAMP,  
    Reason VARCHAR(255),  
    Deletion_Date DATETIME,  
    Note VARCHAR(255),  
    User_ID INT,  
    FOREIGN KEY (User_ID) REFERENCES User(User_ID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Daily_Task (  
    Task_ID_Task INT,  
    User_ID INT,  
    Task_Date DATE, -- Data específica para o conjunto diário de tarefas  
    Name VARCHAR(255) NOT NULL,
```

```
Description VARCHAR(255),  
Status TINYINT, -- Percentual de 0 a 100  
Creation_Date DATETIME DEFAULT CURRENT_TIMESTAMP,  
Completion_Date DATETIME,  
FOREIGN KEY (Task_ID_Task) REFERENCES Task(Task_ID) ON DELETE CASCADE,  
FOREIGN KEY (User_ID) REFERENCES User(User_ID) ON DELETE CASCADE  
);
```

2.2 Popular tabelas com dados de teste

Incluir aqui o script que insere dados nas tabelas cadastrais do sistema

-- Inserir dados de exemplo na tabela User

```
INSERT INTO User (Name, Email, Password, Birth_Date, Gender, Registration_Date,  
Last_Login)
```

```
VALUES ('John Smith', 'john.smith@example.com', 'password123', '1985-05-15  
00:00:00', 'Male', '2024-07-01 10:00:00', '2024-07-01 10:05:00');
```

-- Inserir dados de exemplo na tabela Task

```
INSERT INTO Task (Title, Description, Creation_Date, Reason, Deletion_Date, Note,  
User_ID)
```

```
VALUES ('Task Title', 'Detailed task description', '2024-07-01 10:10:00', 'Reason for the  
task', NULL, 'Note about the task', 1);
```

-- Inserir dados de exemplo na tabela Daily_Task

```
INSERT INTO Daily_Task (Task_ID_Task, User_ID, Task_Date, Name, Description,  
Status, Creation_Date, Completion_Date)  
  
VALUES (1, 1, '2024-07-01', 'Daily Task Name', 'Description of the daily task', 0, '2024-  
07-01 10:20:00', NULL);
```

2.3 Consultas e atualizações da base de dados das principais transações do sistema

Incluir aqui o script com algumas consultas e/ou atualizações que representem as principais transações do sistema. Estas transações são de livre escolha do grupo.

-- Contar o número total de tarefas concluídas por cada usuário

```
SELECT u.Name, COUNT(t.Task_ID) AS total_completed_tasks  
  
FROM User u  
  
LEFT JOIN Task t ON u.User_ID = t.User_ID AND t.Status = 100  
  
GROUP BY u.Name;
```

-- Calcular a média de idade dos usuários cadastrados

```
SELECT AVG(DATEDIFF(CURRENT_DATE, u.Birth_Date) / 365) AS average_age  
  
FROM User u;
```

-- Listar todas as tarefas não concluídas por data de criação

```
SELECT *  
  
FROM Task  
  
WHERE Status < 100  
  
ORDER BY Creation_Date;
```

-- Encontrar o usuário com mais tarefas concluídas

```
SELECT u.Name, COUNT(t.Task_ID) AS total_completed_tasks
FROM User u
LEFT JOIN Task t ON u.User_ID = t.User_ID AND t.Status = 100
GROUP BY u.Name
ORDER BY total_completed_tasks DESC
LIMIT 1;
```

-- Calcular a quantidade média de tarefas diárias criadas por dia

```
SELECT AVG(daily_tasks_count) AS average_daily_tasks_per_day
FROM (
    SELECT COUNT(td.Task_ID_Task) AS daily_tasks_count
    FROM Daily_Task td
    GROUP BY td.Task_Date
) daily_counts;
```

-- Listar usuários que não fizeram login nos últimos 30 dias

```
SELECT *
FROM User
WHERE Last_Login < DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY) OR Last_Login IS
NULL;
```

-- Calcular o número total de tarefas concluídas por mês

```
SELECT MONTH(Completion_Date) AS month, COUNT(*) AS total_completed_tasks
```

```
FROM Task
```

```
WHERE Status = 100
```

```
GROUP BY month;
```

```
-- Listar todas as tarefas diárias associadas a uma tarefa específica
```

```
SELECT td.*
```

```
FROM Task t
```

```
INNER JOIN Daily_Task td ON t.Task_ID = td.Task_ID_Task
```

```
WHERE t.Task_ID = 1;
```

```
-- Contar o número de tarefas diárias concluídas para um determinado hábito
```

```
SELECT COUNT(*)
```

```
FROM Daily_Task
```

```
WHERE Task_ID_Task = 1 AND Completion_Date IS NOT NULL;
```

```
-- Calcular a média de tempo gasto em cada tarefa diária de uma tarefa específica
```

```
SELECT AVG(TIMESTAMPDIFF(SECOND, Creation_Date, Completion_Date)) AS  
average_time_spent_seconds
```

```
FROM Daily_Task
```

```
WHERE Task_ID_Task = 1 AND Completion_Date IS NOT NULL;
```

```
-- Listar todas as diárias não concluídas associadas a tarefas ainda ativas
```

```
SELECT td.*
```

```
FROM Task t
```

```
INNER JOIN Daily_Task td ON t.Task_ID = td.Task_ID_Task
```



```
WHERE t.Status < 100 AND td.Completion_Date IS NULL;
```

```
-- Encontrar a diária mais recentemente criada para um determinado hábito
```

```
SELECT *
```

```
FROM Daily_Task
```

```
WHERE Task_ID_Task = 1
```

```
ORDER BY Creation_Date DESC
```

```
LIMIT 1;
```

```
-- Calcular, para cada hábito, a porcentagem de tarefas diárias concluídas em relação ao total
```

```
SELECT t.Task_ID, COUNT(td.Task_ID_Task) AS total_daily_tasks,
```

```
       SUM(CASE WHEN td.Completion_Date IS NOT NULL THEN 1 ELSE 0 END) AS  
total_completed_daily_tasks,
```

```
       (SUM(CASE WHEN td.Completion_Date IS NOT NULL THEN 1 ELSE 0 END) /  
COUNT(td.Task_ID_Task)) * 100 AS completion_percentage
```

```
FROM Task t
```

```
LEFT JOIN Daily_Task td ON t.Task_ID = td.Task_ID_Task
```

```
GROUP BY t.Task_ID;
```

```
-- Listar todas as tarefas diárias cujos hábitos correspondentes foram excluídos
```

```
SELECT td.*
```

```
FROM Task t
```

```
LEFT JOIN Daily_Task td ON t.Task_ID = td.Task_ID_Task
```

```
WHERE t.Deletion_Date IS NOT NULL;
```

3. ESPECIFICAÇÃO DAS INTERFACES

3.1 Escolher uma transação

Descrever a especificação da interface do sistema da transação escolhida.

Exs: Entrada de Notas Fiscais, Retirada de itens de estoque, transferência bancária

Interface de Cadastro de Usuário :

- Interface Web (Front-end)

- Rota: `/User`

- Método HTTP: `POST`

Campos do Formulário de Cadastro

- Nome: Tipo `VARCHAR(255)`, campo obrigatório (nome completo do usuário)

- Email: Tipo `VARCHAR(255)`, campo obrigatório, formato de e-mail (e-mail do usuário)

- Senha: Tipo `PASSWORD`, campo obrigatório, mínimo de 8 caracteres

- Data de Nascimento: Tipo `DATE`, campo opcional

- Gênero: Tipo `ENUM('Male', 'Female', 'Non-Binary', 'Other')`, campo opcional

- Data de Cadastro: Tipo `DATETIME`, preenchido automaticamente

- Data de Login: Tipo `DATETIME`, preenchido automaticamente com a data do último login

3.2 Escolher outra transação

Interface de Gerenciamento de Tarefas Diárias:

- Interface Web (Front-end)

- Rota: `/Daily_Task`

- Método HTTP: `POST`

****Campos do Formulário para Criação de Tarefas Diárias****

- Nome: Tipo `VARCHAR(255)`, campo obrigatório (nome da tarefa)
- Descrição: Tipo `VARCHAR(255)`, campo obrigatório (descrição da tarefa)
- Status: Tipo `TINYINT` (Percentual de 0 a 100), campo obrigatório
- Data de Criação: Tipo `DATETIME`, preenchido automaticamente
- Data de Conclusão: Tipo `DATETIME`, preenchido automaticamente quando a tarefa é concluída

4. ESPECIFICAÇÃO DAS REGRAS DE NEGÓCIO

4.1 Usar a mesma transação da interface

Regras de Negócio da transação "Cadastro de Usuário":

- Regra 1: O campo email deve ser único no banco de dados.
- Regra 2: A senha deve ter no mínimo 8 caracteres.
- Regra 3: O campo nome é obrigatório e não pode estar vazio.
- Regra 4: Se fornecido, o campo data_nascimento deve estar no formato `YYYY-MM-DD`.
- Regra 5: O campo genero é opcional e pode ter os valores "Male", "Female", "Non-Binary" ou "Other". Se "Other" for selecionado, o usuário deve poder inserir um texto personalizado.

- Regra 6: O campo email deve ter um formato válido de email.
- Regra 7: A senha deve ser armazenada de forma segura, usando técnicas de hash.
- Regra 8: A data de cadastro deve ser preenchida automaticamente com a data atual.
- Regra 9: A data de login deve ser preenchida automaticamente com a data do último login.

4.2 Usar a mesma transação da interface

Regras de Negócio da transação "Tarefas Diárias":

- Regra 1: O nome e a descrição são obrigatórios.
- Regra 2: A data de criação é preenchida automaticamente com a data atual.
- Regra 3: O status inicial deve ser "Pendente".
- Regra 4: A data de conclusão é preenchida automaticamente quando a tarefa é marcada como concluída.