

## *BeachUp*

### – Nome do sistema

*BeachUp*

### – Objetivo do sistema

*Facilitar a organização de partidas e campeonatos informais de beach tennis entre amigos, por meio de nosso aplicativo.*

### – Usuários principais (público-alvo)

- *Participantes de beach tennis amador;*
- *Duplas ou grupos de amigos que jogam com frequência;*
- *Jogadores que desejam organizar ou participar de torneios locais;*
- *Público jovem e adulto, entre 18 e 45 anos.*

### – Principais funcionalidades

- *Cadastro de usuários e times*
- *Agendamento de partidas com data, hora e local.*
- *Convite e confirmação de presença*
- *Registro de resultados das partidas*
- *Chaveamento*
- *Interface Mobile amigável*

### – Problema que será resolvido

*Pessoas sem times formados, ou com integrantes a menos podem enviar convites para times, ou até mesmo ser convidado estando cadastrado em nosso app.*

### – Recursos desejados

*Banco de dados online;  
Interface mobile responsiva;  
Autenticação de usuários;  
Controle de agendamentos e dados persistentes;  
Hospedagem em nuvem;  
Prototipação de telas;*

*Sistema de notificações e lembretes.*

- *Plataforma (Web? Mobile? Desktop?)*

*Nossa plataforma será Mobile.*

- *Prazos e expectativas*

*Semana 1-2: Levantamento de requisitos, definição de escopo.*

*Semana 3-5: Desenvolvimento do backend e início do frontend.*

*Semana 6-8: Integração, testes e ajustes de funcionalidades.*

*Semana 9-10: Entrega final, documentação técnica e apresentação.*

*Tempo previsto de **1 ano**.*

- *Visão geral do projeto*

*Problema: Atualmente, a organização de partidas e campeonatos informais de beach tennis entre amigos, depende de plataformas descentralizadas como grupos de WhatsApp, planilhas ou mensagens diretas. Isso gera confusão, desencontros, perda de histórico e falta de engajamento entre os jogadores.*

*Solução: Desenvolver um aplicativo mobile que centralize todas as etapas da organização esportiva entre duplas ou grupos de beach tennis: criação de times, agendamento de partida, convite, confirmação de jogadores e visualização de histórico e rankings.*

*Usuários e Clientes:*

*Jogadores amadores de beach tennis ( individuais, duplas ou times )*

*Grupos de amigos que se reúnem periodicamente para jogar*

*Recursos principais:*

- Cadastro de usuários e times
- Agendamento de partidas com data, hora e local.
- Convite e confirmação de presença
- Registro de resultados das partidas
- Chaveamento
- Interface Mobile amigável

#### *Canais:*

- Lançamento via Google Play e App Store
- Divulgação orgânica por meio de redes sociais
- Parcerias com quadra de beach tennis
- Grupos de WhatsApp e comunidades esportivas

#### *Custo e Receita:*

##### *Ferramentas e infraestrutura:*

*Firebase- Custo mensal de aproximadamente R\$130,00*

*Domínio - R\$60,00 por ano*

*Google play / App Store: R\$125,00 1 vez só*

*Ferramentas de design ( figma pro ) - 0R\$*

*Marketing inicial: R\$100-300*

#### *Modelo Freemium + Assinatura*

- *Acesso gratuito com recursos básicos*
- *Assinatura mensal (R\$ 9,90 a R\$ 19,90) para:*
  - *Histórico completo de partidas*
  - *Estatísticas e rankings avançados*
  - *Criação de torneios*
  - *Notificações personalizadas*
  - 
  -

### ***Parcerias com quadras***

- *Cobrança de comissão por quadra agendada via app*
- *Cobrança por destaque de quadras no mapa do app*

*Ex: “Quadra Arena do Sol” paga R\$ 49/mês para ser exibida como destaque*

### ***Atores do sistema***

*Jogador – cadastra-se, cria ou entra em um time*

*Time – Entidade que reúne dois ou mais jogadores*

*Sistema- Gerencia dados, rankings, partidas e notificações*

### ***Requisitos Funcionais:***

*O sistema deve permitir o cadastro e login de usuários.*

*O usuário deve poder criar ou entrar em um time*

*O sistema deve permitir o agendamento de partidas com data, hora e local.*

*O sistema deve permitir o agendamento de partidas com data, hora e local*

*O sistema deve permitir convidar outros jogadores para uma partida*

*O sistema deve permitir a confirmação ou recusa de participação nos jogos*

*O sistema deve registrar o placar e resultado da partida*

*O sistema deve exibir partidas futuras agendadas*

*O sistema deve realizar o chaveamento dos times*

*O sistema deve permitir editar o perfil do usuário*

#### *Declaração de requisitos não funcionais*

*O sistema deve ser desenvolvido com arquitetura mobile (Native React).*

*O sistema deve armazenar os dados em nuvem.*

*O sistema deve possuir uma interface responsiva e intuitiva.*

*O sistema deve garantir a integridade dos dados de login e partidas.*

*O sistema deve funcionar em dispositivos Android e IOS.*

*O tempo de resposta da aplicação deve ser inferior a 2 segundos por ação.*

*O Sistema deve permitir múltiplos acessos simultâneos.*

---

### *Casos de uso*

#### *Caso de uso 0-*

Ator Principal: Usuário novo

Pré-condições: O usuário não possui conta cadastrada.

Fluxo Principal:

1. O usuário seleciona a opção “Cadastrar” na tela inicial.
2. O sistema exibe o formulário de cadastro com campos obrigatórios (nome, e-mail, senha, etc.).
3. O usuário preenche as informações solicitadas.
4. O usuário confirma o cadastro.
5. O sistema valida os dados, cria a conta e exibe mensagem de sucesso.

Fluxo Alternativo:

– No passo 3, se o e-mail já estiver cadastrado ou algum campo obrigatório estiver em branco, o sistema exibe mensagens de erro e impede o cadastro até a correção.

Pós-condições: O usuário está registrado e pode acessar o sistema após autenticação

## Caso de Uso 1: Cadastrar Partida

Ator Principal: Jogador (organizador)

Pré-condições: O jogador está autenticado.

Fluxo Principal:

1. O jogador seleciona a opção “Cadastrar Partida” no menu.
2. O sistema exibe um formulário de cadastro da partida.
3. O jogador preenche data, hora, local, formato (duplas/simples) e número de vagas.
4. O jogador confirma o cadastro.
5. O sistema armazena as informações e gera um identificador único da partida.
6. O sistema exibe mensagem de sucesso.

Fluxo Alternativo:

– No passo 3, se houver campos obrigatórios em branco ou data/hora inválidas, o sistema exibe mensagens de erro e impede o cadastro até correção.

Pós-condições: A partida é cadastrada e aparece na lista de partidas agendadas do organizador.

## Caso de Uso 2: Convidar Jogadores

Ator Principal: Jogador (organizador)

Pré-condições: Existe uma partida cadastrada pelo organizador.

Fluxo Principal:

1. O organizador acessa a página da partida.

2. O organizador seleciona “Convidar Jogadores”.
3. O sistema exibe lista de contatos/jogadores e campo para buscar por usuário.
4. O organizador seleciona os convidados e envia os convites.
5. O sistema registra os convites e envia notificações.

Fluxo Alternativo:

– No passo 4, se a partida já estiver com todas as vagas preenchidas, o sistema bloqueia novos convites e informa indisponibilidade.

Pós-condições: Convites ficam pendentes até confirmação/recusa dos jogadores.

### **Caso de Uso 3: Confirmar Presença**

Ator Principal: Jogador convidado

Pré-condições: Jogador autenticado com convite pendente.

Fluxo Principal:

1. O jogador acessa “Convites” ou a notificação recebida.
2. O jogador abre os detalhes da partida.
3. O jogador seleciona “Confirmar Presença” ou “Recusar”.
4. O sistema registra a resposta e atualiza a lista de participantes.
5. O sistema exibe mensagem de confirmação.

Fluxo Alternativo:

– No passo 3, se as vagas forem preenchidas antes da confirmação, o sistema informa lotação esgotada e oferece entrar em lista de espera.

Pós-condições: A presença do jogador fica registrada para a partida (ou ele entra na lista de espera/recusa).



### **Casos de uso 4: Registrar resultado da partida**

Ator Principal: Jogador organizador

Pré-condições: Partida realizada e organizador autenticado.

Fluxo Principal:

1. O organizador acessa “Histórico/Partidas Finalizadas”.
2. O organizador seleciona a partida.
3. O organizador clica “Registrar Resultado”.
4. O sistema exibe formulário para placares por set e vencedores.
5. O organizador confirma o registro.
6. O sistema salva o resultado e atualiza chaveamento.

Fluxo Alternativo:

– No passo 4, se o placar for inconsistente (ex.: set com pontuação inválida), o sistema exibe erro e solicita correção.

## **Caso de uso 5: Gerenciar time**

Ator Principal: Jogador (capitão do time)

Pré-condições: Capitão autenticado e associado a um time.

Fluxo Principal:

1. O capitão acessa “Meu Time”.
2. O capitão seleciona “Editar Time”.
3. O sistema exibe opções para alterar nome, foto e membros.
4. O capitão adiciona/remove membros e salva alterações.
5. O sistema atualiza os dados do time e confirma sucesso.

Fluxo Alternativo:

– No passo 4, se o jogador convidado não aceitar entrar no time, o sistema mantém status pendente e notifica o capitão.

Pós-condições: O cadastro do time é atualizado; convites pendentes ficam aguardando resposta.

## **Caso de uso 6: Consultar agenda e histórico**

Ator Principal: Jogador

Pré-condições: Jogador autenticado.

Fluxo Principal:

1. O jogador acessa “Agenda”.
2. O sistema exibe partidas futuras com data, hora, local e status (confirmado/pendente).
3. O jogador pode filtrar por meu time, local ou data.
4. O jogador alterna para “Histórico” para ver partidas passadas com resultados.
5. O sistema apresenta detalhes da partida selecionada.

Fluxo Alternativo:

– No passo 2, se não houver partidas futuras/anteriores, o sistema exibe mensagem “Nenhuma partida encontrada” e sugere “Cadastrar Partida”.

Pós-condições: Nenhuma mudança de estado é obrigatória; o jogador apenas consulta informações.