

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Painel Administrativo para o Piloto
BikeSP**

TCC IME/USP

Mikhael Abdallah de Oliveira Pinto

MONOGRAFIA FINAL

MAC 499 — TRABALHO DE
FORMATURA SUPERVISIONADO

Supervisor: Prof. Dr. Fabio Kon
Cossupervisor: Prof. Dr. Paulo Roberto Miranda Meirelles

São Paulo
2025

*O conteúdo deste trabalho é publicado sob a licença CC BY 4.0
(Creative Commons Attribution 4.0 International License)*

*Esta seção é opcional e fica numa página separada;
ela pode ser usada para uma dedicatória ou epígrafe.*

[illegible]

Resumo

Mikhael Abdallah de Oliveira Pinto. **Painel Administrativo para o Piloto BikeSP: TCC IME/USP**. Monografia (Bacharelado). Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2025.

Este Trabalho apresenta o desenvolvimento de um painel administrativo para o piloto do programa Bike SP, iniciativa da Prefeitura de São Paulo em parceria com a USP voltada à promoção do ciclismo urbano por meio de incentivos financeiros. O trabalho descreve o contexto e implementação do painel, que apoiou o pré-teste entre Abril e Maio, a inscrição de 3 mil pessoas e o acompanhamento do projeto piloto, que teve duração de quatro semanas. O painel foi essencial para a gestão operacional do projeto, sendo importante para a gestão de cadastros, acompanhamento do uso do app, monitoramento das viagens e gestão dos pagamentos.

Palavras-chave: BikeSP. painel administrativo. incentivos financeiros. mobilidade urbana.

Abstract

Mikhael Abdallah de Oliveira Pinto. **Administrative Dashboard for the BikeSP Pilot: Undergraduate Capstone.** Capstone Project Report (Bachelor). Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2025.

This Thesis presents the development of an administrative dashboard for the Bike-SP program pilot, an initiative by the São Paulo City Hall in partnership with USP to promote urban cycling through financial incentives. The work describes the context and implementation of the dashboard, which supported a pre-test (April-May), the registration of 3,000 users, and the monitoring of the four-week pilot project. The dashboard was essential for the project's operational management, handling user registration, app usage tracking, trip monitoring, and payment management.

Keywords: BikeSP. administrative dashboard. financial incentives. urban mobility.

Lista de abreviaturas

API	Interface de Programação de Aplicações (<i>Application Programming Interface</i>)
IME	Instituto de Matemática e Estatística
USP	Universidade de São Paulo

Lista de símbolos

Lista de figuras

3.1	Diagrama simplificado dos componentes do sistema BikeSP.	8
3.2	Interface de inserção de usuários no painel administrativo, com botão para executar o script de importação do LimeSurvey (dados fictícios).	10
3.3	Interface de listagem de usuários com funcionalidades de busca, ordenação e paginação (dados fictícios).	12
3.4	Interface de gerenciamento de coortes e visualização de quantidade de participantes por grupo (dados fictícios).	14
3.5	Interface de filtragem de participantes por coorte (dados fictícios).	15
3.6	Interface de upload de atribuição de coortes em lote (dados fictícios).	16
3.7	Interface de exportação de coortes (dados fictícios).	16
3.8	Formulário de criação de bônus no painel administrativo (dados fictícios).	18
3.9	Formulário de atribuição de bônus aos participantes (dados fictícios).	19
3.10	Listagem de bônus cadastrados com funcionalidades de busca, paginação e ativação/desativação (dados fictícios).	20
3.11	Interface de geração de lista de créditos a pagar via SPTrans (dados fictícios).	22
3.12	Interface de upload de comprovante de pagamentos realizados pela SPTrans (dados fictícios).	23
3.13	Interface de listagem de contestações com busca e paginação (dados fictícios).	25
3.14	Interface de aprovação/rejeição de contestação (dados fictícios).	26
3.15	Interface de listagem de viagens com funcionalidades de busca, ordenação e paginação (dados fictícios).	27
3.16	Visualização de trajeto de viagem no mapa interativo com marcadores de pontos GPS (dados fictícios).	28
3.17	Formulário de criação e envio de notificações para participantes (dados fictícios).	30

Lista de tabelas

Lista de programas

Sumário

1	Introdução	1
2	O Piloto BikeSP e seu painel administrativo	3
2.1	Programa Bike SP	3
2.2	Piloto do BikeSP	4
2.3	Objetivos do trabalho	5
3	O painel administrativo	7
3.1	Arquitetura de software	7
3.2	Processo de desenvolvimento	8
3.3	Abas do painel e uso no piloto	9
3.3.1	Inserção de usuários	9
3.3.2	Usuários	11
3.3.3	Coortes	13
3.3.4	Bônus	17
3.3.5	Remuneração	20
3.3.6	Contestações	23
3.3.7	Viagens	26
3.3.8	Notificações	29
3.3.9	Localizações	31
4	Resultados e avaliação	33
4.1	Resultados atingidos	33
4.2	Análise	34
5	Conclusão	35
	Referências	37

Capítulo 1

Introdução

A bicicleta é um meio de transporte sustentável que pode contribuir para a melhoria da qualidade de vida urbana em diversas dimensões, como redução da poluição ambiental, melhora da saúde pública e diminuição dos custos sociais e individuais do transporte. Nesse sentido, políticas públicas que incentivam o uso da bicicleta são cada vez mais importantes para o planejamento de transporte urbano (SHELLER, 2018; JEEKEL, 2017). Em São Paulo, maior cidade do Brasil com mais de 12 milhões de habitantes e 24 milhões de viagens diárias, apenas 0,9% de todas as viagens eram feitas por bicicleta em 2017 (METRÔ SÃO PAULO, 2017). Recentemente, a Prefeitura estabeleceu a meta de elevar a participação modal do ciclismo para 3,2% até 2028 (PREFEITURA DE SÃO PAULO, 2022).

O programa Bike SP tem origem na **Lei Municipal 16.547/2016** (PREFEITURA DE SÃO PAULO, 2016), que prevê a concessão de créditos de mobilidade a cidadãos que realizam deslocamentos utilizando bicicleta. A lei especifica que esses créditos serão dados aos usuários registrados no sistema *Bilhete Único*, podendo ser utilizados para pagamento de ônibus, metrô e trem. No entanto, a implementação foi adiada devido à ausência de decreto regulamentar que especifique aspectos práticos da política, como critérios de elegibilidade, valor do benefício e método de medição das distâncias percorridas.

Incentivos financeiros para ciclistas já existem em alguns países, principalmente na Europa, como na Holanda (subsídio de €0,25 por quilômetro) e na Bélgica (TENNANT, 2022; THE BRUSSELS TIMES WITH BELGA, 2022). Entretanto, devido à escassez de implementações anteriores em contextos semelhantes ao de São Paulo e de evidências científicas que orientem seu desenho e execução, decidiu-se fornecer suporte científico à política pública através da implementação de um projeto piloto (LIMA *et al.*, 2023; PEARSON *et al.*, 2020).

A partir de 2023, o IME-USP iniciou o projeto de pesquisa em parceria com o grupo INCT InterSCity, com o objetivo de elaborar um experimento controlado capaz de gerar evidências empíricas para guiar a regulamentação efetiva desta lei. Este estudo piloto foi desenhado e conduzido por pesquisadores do IME-USP e instituições parceiras, incluindo especialistas em mobilidade cicloviária e a *Eindhoven University of Technology* (TU/e).

O presente trabalho, desenvolvido em 2025, buscou viabilizar a execução do experimento através de uma equipe de desenvolvimento e suporte composta por 6 pessoas do

grupo de pesquisa, projetando e implementando uma versão do sistema com funcionalidades estendidas para atender às demandas operacionais e de pesquisa do piloto.

O painel administrativo desenvolvido neste trabalho tem como propósito apoiar a operação do piloto e sua avaliação, oferecendo: (i) visibilidade sobre cadastros, elegibilidade e amostras; (ii) acompanhamento de viagens registradas no aplicativo móvel; (iii) cálculo e auditoria de créditos; e (iv) insumos para monitoramento e análise de dados.

Este trabalho está organizado da seguinte forma: o Capítulo 2 apresenta o contexto do Piloto BikeSP, descrevendo o programa Bike SP, o desenho experimental do piloto com suas coortes e cronograma, e os objetivos do trabalho; o Capítulo 3 detalha a implementação do painel administrativo, incluindo a arquitetura de software e as funcionalidades desenvolvidas para cada aba do sistema; o Capítulo 4 apresenta os resultados alcançados durante a execução do piloto e uma análise dos impactos do painel nas operações; por fim, o Capítulo 5 conclui o trabalho e discute trabalhos futuros.

Capítulo 2

O Piloto BikeSP e seu painel administrativo

Este capítulo apresenta o contexto do programa Bike SP e do projeto piloto que foi desenvolvido para gerar evidências científicas para sua regulamentação. A Seção 2.1 descreve o programa Bike SP e sua origem na Lei Municipal 16.547/2016. A Seção 2.2 detalha o piloto experimental, incluindo cronograma, processo de inscrições e o desenho experimental com três coortes. Por fim, a Seção 2.3 apresenta os objetivos específicos deste trabalho.

2.1 Programa Bike SP

O Bike SP é um programa público que visa incentivar deslocamentos funcionais por bicicleta por meio de créditos de mobilidade. Já o *piloto* é uma etapa experimental, delimitada no tempo e no escopo, concebida para testar variáveis de implementação (por exemplo, níveis de incentivo), observar comportamentos de uso e coletar evidências que subsidiem o aperfeiçoamento e o eventual escalonamento do programa. Em outras palavras, o programa é a política pública; o piloto é o experimento que informa sua execução (LIMA *et al.*, 2023; INCT INTERSCITY, 2025).

Um projeto piloto é um experimento de pequena escala que permite testar a viabilidade e efetividade de uma intervenção antes de escaloná-la para uma população maior (PEARSON *et al.*, 2020). A escolha do piloto como meio de guiar a implementação do Bike SP é motivada pela necessidade de dados precisos e do mundo real.

Métodos tradicionais de pesquisa por questionários, embora úteis, frequentemente falham em capturar a complexidade do comportamento real, pois pode haver divergência significativa entre o que indivíduos afirmam que farão e o que efetivamente fazem (RUNDLE-THIELE, 2009). Assim, o piloto simula a política real ao compensar participantes por suas viagens de bicicleta, fornecendo dados sobre como tal política poderia funcionar na prática.

Além disso, o programa Bike SP tem potencial para integrar as agendas de justiça social e ambiental, ao reduzir os custos de mobilidade de grupos de menor renda e aumentar a

equidade no transporte. Os créditos de transporte público obtidos pedalando podem ser usados para pagar outras viagens (por ônibus, metrô ou trem), beneficiando especialmente populações de baixa renda.

Dessa forma, o programa promove não apenas o aumento da participação modal do ciclismo, mas também a equidade de mobilidade(VANOUTRIVE e COOPER, 2020; JEEKEL, 2017).

2.2 Piloto do BikeSP

O piloto do BikeSP foi executado entre abril e agosto, envolvendo diferentes fases desde o pré-teste até a execução principal do experimento, com participação de 1217 ciclistas selecionados. A seguir, descrevemos o cronograma completo, o processo de inscrições e seleção, e o desenho experimental com suas três coortes.

Cronograma.

- **Abril–Maio:** pré-teste com participantes beta; o painel foi utilizado desde o início para acompanhar registros e validar fluxos.
- **Três meses de inscrições:** processo conduzido via LimeSurvey; ao final, foram recebidas **cerca de 3000 candidaturas**.
- **Julho:** uso ampliado do aplicativo; a equipe operou o painel para monitorar o uso e oferecer suporte a usuários com dificuldades.
- **Agosto:** execução do piloto principal em **4 semanas**.

O formulário de inscrição criado na plataforma LimeSurvey permitiu coleta padronizada de dados demográficos e de contexto de mobilidade. Com o apoio de aparições na mídia, foram obtidas aproximadamente **3 mil inscrições**. A seleção final considerou o desenho experimental e o balanceamento de perfis e regiões, resultando em **1217 candidatos selecionados** para participação no piloto. Para evitar viés, foram excluídos cicloativistas reconhecidos, priorizando usuários representativos do público-alvo da política pública.

O experimento foi estruturado com base em **três coortes**, que são grupos experimentais com diferentes níveis de remuneração, conforme planejado pela equipe de economistas:

- **Coorte 1 (controle):** sem remuneração por viagens;
- **Coorte 2:** remuneração de R\$ 0,30 por quilômetro pedalado, limitado a 2 viagens por dia;
- **Coorte 3:** remuneração de R\$ 0,60 por quilômetro pedalado, também limitado a 2 viagens por dia.

A título de exemplo, um participante da Coorte 3 que percorresse 8 km em uma viagem receberia R\$ 4,80 — valor próximo ao de uma passagem de ônibus na época. Importante notar que **não houve remuneração nos finais de semana**, permitindo observar o comportamento de uso do aplicativo sem o incentivo financeiro. Ao longo das quatro semanas do piloto, os participantes tiveram suas coortes alternadas semanalmente, conforme desenho conduzido pelos economistas, de modo a permitir a avaliação causal do efeito dos diferentes níveis de incentivo.

2.3 Objetivos do trabalho

Este trabalho teve como objetivo desenvolver um **painel administrativo** capaz de dar suporte efetivo aos processos fundamentais do programa durante o piloto, atuando como ferramenta central de operação e governança. Em alto nível, o painel deve oferecer: (i) visibilidade sobre cadastros, elegibilidade e amostras; (ii) acompanhamento de viagens registradas no aplicativo; (iii) cálculo de créditos com trilhas de auditoria e apoio a pagamentos; (iv) tratamento de contestações; e (v) exportações e insumos para monitoramento e análise.

O próximo capítulo apresenta detalhadamente a implementação do painel administrativo, descrevendo a arquitetura de software do sistema BikeSP e cada uma das funcionalidades desenvolvidas para atender aos objetivos aqui descritos.

Capítulo 3

O painel administrativo

Este capítulo descreve a implementação do painel administrativo desenvolvido para apoiar o piloto BikeSP. A Seção 3.1 apresenta a arquitetura de software do sistema completo, mostrando como o painel se integra aos demais componentes. A Seção 3.2 descreve o processo de desenvolvimento adotado pela equipe. A Seção 3.3 detalha cada uma das funcionalidades implementadas nas diferentes abas do painel, incluindo inserção de usuários, gerenciamento de coortes, bônus, remuneração, contestações, viagens, notificações e localizações.

3.1 Arquitetura de software

O sistema BikeSP é composto por múltiplos componentes distribuídos em diferentes infraestruturas, conforme ilustrado na Figura 3.1. O **Painel Administrativo** atua como componente central de orquestração, integrando-se aos demais módulos do sistema.

Os usuários realizavam inscrição através do **Formulário LimeSurvey**, hospedado em uma máquina virtual NuvemUSP. Após o período de inscrições, um administrador utiliza o painel para executar um *script* que extraía os dados dos candidatos do LimeSurvey e os inseria no banco de dados PostgreSQL do BikeSP, já filtrando os candidatos selecionados pela equipe de pesquisa pelos critérios de elegibilidade e balanceamento experimental definidos.

A máquina virtual Amazon hospeda o **Banco de Dados PostgreSQL** e o **Back-end Flask**, que implementa a lógica de negócio do sistema (validação de viagens, cálculo de créditos, gerenciamento de usuários e coortes). O painel comunica-se com o back-end via APIs REST para consultar e manipular dados.

O **App Android** é utilizado pelos participantes para registrar viagens de bicicleta. As viagens são enviadas ao back-end, que as valida e processa. O painel permite que administradores visualizem, auditem e, se necessário, corrijam dados de viagens (por exemplo, erros de geolocalização).

O sistema integra-se com: (i) **Google Firebase** para autenticação e notificações push; (ii) **APIs de Geolocalização** (TomTom e Nominatim) para validação e correção de co-

ordenadas GPS; e (iii) **Loja Virtual SPTrans** para processamento de pagamentos dos créditos acumulados pelos participantes.

O painel é acessado pela **equipe de suporte** e operação, permitindo gestão de cadastros, atribuição de coortes, criação de bônus, aprovação de contestações e geração de relatórios para a equipe de economistas.

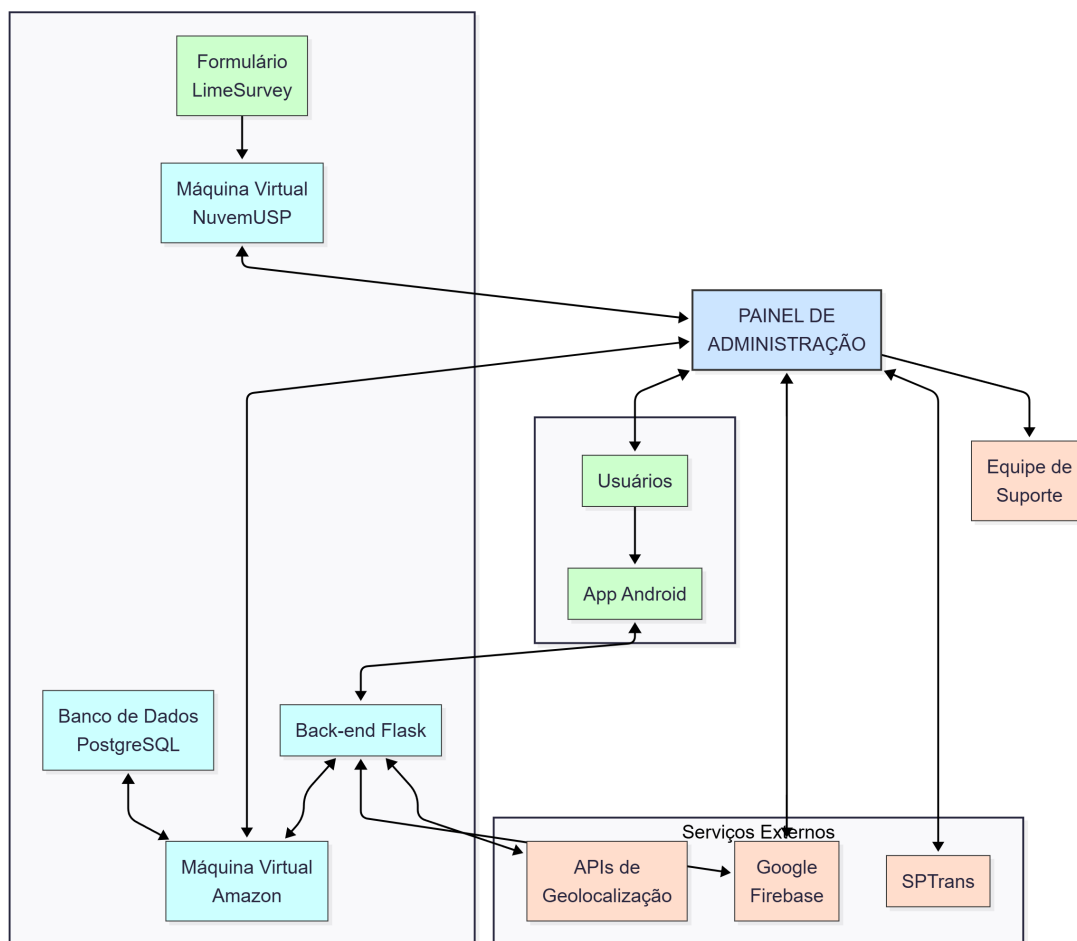


Figura 3.1: Diagrama simplificado dos componentes do sistema BikeSP.

3.2 Processo de desenvolvimento

O desenvolvimento envolveu múltiplos stakeholders e ciclos curtos de validação: economistas (por exemplo, Thainá), suporte/atendimento (Sávio), coordenação e orientação (Fábio e Paulo), além da equipe técnica. As demandas foram priorizadas e acompanhadas via registro de *issues*, criação de *pull requests* e revisão colaborativa, com homologação contínua durante o pré-teste, o período de inscrições, a preparação de julho e a execução do piloto.

3.3 Abas do painel e uso no piloto

3.3.1 Inserção de usuários

O processo de recrutamento de participantes para o piloto BikeSP iniciou-se com ampla divulgação midiática (entrevistas em jornais, redes sociais, site institucional), direcionando interessados para formulário online hospedado em instância LimeSurvey na NuvemUSP. O formulário coletava: dados pessoais (nome, CPF, email, data de nascimento, gênero), número do Bilhete Único (pré-requisito para participação, pois remuneração seria creditada neste cartão), senha para acesso ao aplicativo móvel, endereços textuais de até três localizações (tipicamente casa e trabalho/escola), telefone para contato, e concordância com Termo de Consentimento Livre e Esclarecido (TCLE — documento obrigatório aprovado por Comitê de Ética para pesquisas envolvendo seres humanos). Durante o período de inscrições, o formulário recebeu aproximadamente 3.000 submissões, volume que requeria automação para transferência ao banco de dados PostgreSQL do sistema BikeSP.

A funcionalidade de inserção em lote, acessível via botão “Inserir Usuários!” no painel administrativo, executa script que: (i) conecta à API do LimeSurvey e baixa todas respostas ainda não processadas; (ii) itera sobre cada resposta aplicando validações (CPF válido via algoritmo de verificação de dígitos, email em formato correto e único no sistema, Bilhete Único com 15 dígitos numéricos, senha com mínimo de caracteres configurável); (iii) criptografa a senha antes de armazenar (segurança crítica); (iv) geocodifica endereços textuais chamando APIs externas (TomTom, Google Maps, Nominatim OSM) para converter em coordenadas GPS; (v) armazena dados cadastrais da pessoa, vinculação do cartão Bilhete Único com status ativo, e coordenadas de casa e trabalho; (vi) atribui participante a coorte inicial (pode ser randomizado ou seguir configuração padrão); e (vii) registra logs detalhados de sucesso/falha para cada usuário processado.

Execução ocorre em thread separada (processamento assíncrono) para não bloquear interface administrativa durante operação (geocodificação via APIs externas é gargalo principal). Botão fica desabilitado durante processamento, prevenindo cliques duplos que iniciariam múltiplas execuções simultâneas. Mensagem inicial “Usuários novos estão sendo inseridos!” confirma início; mensagem final reporta quantidade inserida (“X usuários inseridos com sucesso”) ou erros encontrados. A Figura 3.2 mostra a interface desta funcionalidade (dados fictícios para fins ilustrativos).



Figura 3.2: Interface de inserção de usuários no painel administrativo, com botão para executar o script de importação do LimeSurvey (dados fictícios).

O sistema implementa verificação de CPF existente antes de tentar inserção, prevenindo duplicações mesmo se script executado múltiplas vezes sobre mesmas respostas do LimeSurvey (comportamento idempotente). Quando endereço falha em geocodificação (10-20% dos casos conforme discussão na subseção de Localizações), localização é criada marcada como “inválida”, permitindo participante existir no sistema mas impedindo uso daquela localização em viagens até correção manual via interface administrativa. Esta estratégia parcialmente otimista (criar usuário mesmo com dados incompletos) prioriza não bloquear participação — melhor ter usuário com localização a corrigir que rejeitar inscrição completamente. Mensagens de aviso (“Y usuários faltaram a serem inseridos”, “Z endereços não puderam ser geocodificados”) orientam reexecução do script (para tentar novamente usuários que falharam transientemente) e correção manual de localizações.

Embora o formulário LimeSurvey coletasse aproximadamente 3.000 inscrições, apenas 1.217 participantes foram efetivamente inseridos no sistema e incluídos no piloto. Critérios de elegibilidade aplicados durante seleção (manualmente por economistas antes de liberar CPFs para script de inserção) incluíam: residência em São Paulo (verificação via CEP), posse de Bilhete Único ativo, não ser cicloativista regular (auto-declaração — para evitar viés de seleção, experimento buscava medir impacto em usuários representativos, não entusiastas pré-dispostos), e disponibilidade para participar durante seis meses. Preferência adicional para diversidade de perfis socioeconômicos e geográficos. Esta separação entre coleta ampla pelo LimeSurvey e inserção seletiva pelo script permitiu flexibilidade experimental mantendo integridade metodológica.

A primeira execução do script, inserindo 1.200 usuários simultaneamente, foi coordenada com atribuição inicial de coortes via funcionalidade de upload em lote (descrita na subseção de Coortes). Workflow combinado: economistas geravam CSV com (CPF, ID_Coorte) aplicando randomização estratificada; administrador executava script de inserção LimeSurvey→PostgreSQL; aguardava confirmação de sucesso; imediatamente executava upload de atribuição de coortes; verificava balanceamento (400 usuários por coorte); enviava notificações de boas-vindas via sistema de notificações. Este processo

único e crítico ocorreu na véspera do início oficial do piloto, requiring coordenação temporal precisa para permitir resolução de problemas antes que participantes recebessem instruções para ativar o aplicativo.

Este processo de inserção em lote via script administrativo foi desenvolvido exclusivamente para o projeto piloto, onde era necessário selecionar manualmente 1.217 participantes dentre 3.000 inscritos aplicando critérios de elegibilidade (residência em São Paulo, não ser cicloativista, diversidade socioeconômica). O formulário LimeSurvey serviu como porta de entrada ampla para candidatos, enquanto o script de inserção permitia controle fino sobre quem seria efetivamente incluído no experimento controlado.

Para a implementação municipal em larga escala (10.000-50.000 usuários), quando o programa BikeSP for liberado para todos os ciclistas de São Paulo, esta arquitetura de duas etapas (LimeSurvey → script → banco de dados) seria substituída por cadastro direto e automático via aplicativo móvel. Usuários interessados instalariam o aplicativo, preencheriam dados cadastrais diretamente na interface, e seriam automaticamente integrados ao sistema sem necessidade de aprovação ou processamento em lote por administradores. O formulário LimeSurvey e o script de inserção foram ferramentas necessárias apenas para o contexto experimental do piloto, não fazendo parte da arquitetura planejada para a política pública permanente.

3.3.2 Usuários

Durante o piloto BikeSP, com aproximadamente 1.200 participantes selecionados dentre 3.000 inscritos, a equipe de suporte enfrentava demandas diárias de atendimento. Questões como “não consigo acessar o aplicativo”, “minha viagem não foi computada”, “como altero meu endereço cadastrado” requeriam identificação rápida do participante no sistema para diagnóstico e resolução. A interface de gerenciamento de usuários tornou-se, portanto, o ponto de partida no sistema administrativo para praticamente todas as operações de suporte, auditoria e análise.

A tela de usuários apresenta tabela com seis colunas principais: ID (identificador único), nome completo, email, CPF, número do Bilhete Único, e indicador mostrando se o participante completou o cadastro de senha (permitindo acesso ao aplicativo). O sistema consulta dados pessoais vinculados aos cartões Bilhete Único ativos, utilizando técnicas de otimização para obter contagem total sem sobrecarregar o banco de dados — importante para responsividade em conjuntos de dados com milhares de registros. A coluna “Tem Senha” verifica se existe senha cadastrada para aquele usuário, indicador crucial para diagnosticar problemas de acesso (usuário inscrito mas sem senha configurada não consegue entrar no aplicativo).

O sistema oferece busca unificada por nome (parcial, *case-insensitive*) ou ID exato. Por exemplo, buscar “silva” retorna “Silva João”, “Silvana Costa”, “Marcos da Silva”, enquanto buscar “123” retorna apenas o participante com ID exatamente 123. Esta abordagem dual (nome parcial vs. ID exato) equilibra flexibilidade para buscas exploratórias (quando administrador conhece apenas nome aproximado) e precisão para operações dirigidas (quando ID já foi identificado em interação anterior ou viagem específica). Importante notar que a busca não cobre email, CPF ou Bilhete Único diretamente; para localizar por

estes atributos, utiliza-se ordenação combinada com inspeção visual. A Figura 3.3 apresenta a interface de listagem (dados fictícios para fins ilustrativos).

Pessoas

Ir para página ↕

Tamanho da Página

ID ↑	Nome	E-mail	CPF	Bilhete Único	Tem Senha
1	Usuário fictício 1	1@gmail.com	11111111111	11111	Sim
2	Usuário fictício 2	2@gmail.com	22222222222	22222	Sim
3	Usuário fictício 3	3@gmail.com	33333333333	33333	Não
4	Usuário fictício 4	4@gmail.com	44444444444	44444	Não
5	Usuário fictício 5	5@gmail.com	55555555555	55555	Não

Página 1 de 4

ANTERIOR

PRÓXIMO

Figura 3.3: Interface de listagem de usuários com funcionalidades de busca, ordenação e paginação (dados fictícios).

Cada cabeçalho de coluna funciona como botão de ordenação: primeiro clique ordena ascendentemente, segundo clique inverte para descendente, com indicadores visuais (setas ↑↓) mostrando coluna ativa e direção. Ordenação por nome facilita localização alfabética; por ID permite identificar usuários cadastrados cronologicamente (IDs crescentes); por “Tem Senha” agrupa participantes que completaram vs. não completaram ativação.

A interface oferece quatro densidades: 5, 10, 15 (padrão) ou 20 usuários por página, selecionáveis via dropdown. Navegação ocorre por botões “Anterior”/“Próximo” (desabilitados nos extremos) ou campo de entrada direta “Ir para página X”, validando automaticamente se página solicitada existe. Indicador “Página X de Y” fornece senso de escala do conjunto de dados. Esta flexibilidade mostrou-se útil em diferentes contextos operacionais: densidade baixa (5-10 itens) para inspeção cuidadosa com múltiplas abas abertas; densidade alta (20 itens) para varredura rápida ou busca semi-manual quando termo de busca textual não é aplicável.

A tela de usuários serve como ponto de entrada para diversos fluxos operacionais críticos durante o piloto. *Cenário 1: Suporte via email* — participante reporta viagem não creditada; administrador busca por nome, identifica ID, navega à tela de viagens filtrando por aquele ID, localiza viagem em questão e diagnostica problema (ex: trajeto fora dos locais cadastrados). *Cenário 2: Problema de acesso ao app* — participante não consegue fazer login; administrador busca por nome, verifica coluna “Tem Senha”; se “Não”, orienta participante a completar cadastro via funcionalidade de redefinição de senha. *Cenário 3: Ajuste de localização* — participante mudou de endereço; administrador identifica ID, navega à tela de localizações, atualiza coordenadas e endereço textual. *Cenário 4: Contestação de viagem*

— administrador recebe disputa; identifica usuário, obtém ID, acessa tela de contestações para processar aprovação/rejeição. Em todos estes cenários, a rapidez da busca e clareza da informação apresentada determinam eficiência do atendimento.

Durante o período de piloto, a tela de usuários foi acessada diariamente pela equipe de suporte. A métrica “Tem Senha” permitiu identificar participantes que não completaram ativação do aplicativo, orientando campanhas de engajamento. Recursos como ordenação e paginação reduziram significativamente a dependência de consultas manuais ao banco de dados, tornando as operações de suporte de primeiro nível mais ágeis e acessíveis para toda a equipe.

3.3.3 Coortes

O piloto BikeSP foi estruturado como experimento controlado randomizado (RCT — *Randomized Controlled Trial*) visando responder questão de implementação fundamental (IQ1): “Como o número de viagens de bicicleta varia com diferentes valores de remuneração?” Esta metodologia permite isolar efeito de incentivos financeiros de fatores confundidores, fornecendo evidências causais robustas para regulamentação da Lei Municipal 16.547/2016. Participantes foram alocados a três coortes (grupos experimentais) com diferentes níveis de remuneração por quilômetro pedalado: Coorte 0 (controle, sem remuneração), Coorte 1 (incentivo baixo, R\$ 0,30/km, equivalente a 1/16 da tarifa de transporte público), e Coorte 2 (incentivo alto, R\$ 0,60/km, equivalente a 1/8 da tarifa). A escolha destes valores baseou-se em estudo do Banco Mundial sobre elasticidade-preço de demanda por ciclismo em São Paulo, onde remunerações nesta faixa mostraram impacto significativo sem comprometer viabilidade orçamentária.

Característica distintiva do desenho foi rotação semanal de coortes: cada participante alternava entre as três condições a cada semana, experimentando todas coortes ao longo do piloto. Esta estratégia reduz viés de seleção (todos expostos a todas condições) e aumenta poder estatístico para detectar diferenças intra-indivíduos (cada participante serve como próprio controle). Adicionalmente, remuneração foi desativada em finais de semana para todas coortes, permitindo observar retenção de comportamento na ausência de incentivo financeiro — descoberta notável foi que 9 das 10 viagens mais longas ocorreram justamente nestes períodos sem remuneração, evidência de internalização do hábito de pedalar.

A interface de coortes oferece funcionalidades para criação, listagem, e atribuição de participantes a grupos. Criação/atualização requer ID numérico único (1, 2, 3) e valor de remuneração por quilômetro; sistema automaticamente cria nova coorte se ID inexistente, ou atualiza valor se ID já cadastrado (útil para ajustes durante piloto sem recriar grupo). Listagem exibe ID, remuneração formatada como moeda (R\$ 0,30), e quantidade de membros associados, com busca por ID específico e paginação (5 itens/página). Durante piloto com 1.217 participantes, distribuição manteve-se aproximadamente balanceada: 400 participantes por coorte em qualquer semana dada, essencial para comparabilidade estatística. A Figura 3.4 mostra esta interface (dados fictícios para fins ilustrativos).

Criar/Atualizar Coorte

Utilize o formulário abaixo para criar/atualizar uma coorte:

ID da Coorte

ex.: 1

Remuneração

ex.: 10.50

CRIAR/ATUALIZAR COORTE

Coortes Existentes

Buscar por ID...

ID	Remuneração	Quantidade de Membros
3	R\$ 0,60	406
2	R\$ 0,30	406
1	R\$ 0,00	405

Página 1

ANTERIOR

PRÓXIMO

Figura 3.4: Interface de gerenciamento de coortes e visualização de quantidade de participantes por grupo (dados fictícios).

A tela dedicada lista todos usuários com respectivas coortes, exibindo ID, nome, email, total de viagens realizadas, e ID da coorte atual. Sistema de filtros combinados permite: busca por ID de pessoa, nome parcial (*case-insensitive*), email, ID de coorte específica, e intervalo de viagens (mínimo/máximo). Filtros aplicam-se simultaneamente via lógica AND e persistem durante paginação. Esta funcionalidade atendeu múltiplos casos de uso: (i) verificar balanceamento de grupos (filtrar por ID de coorte, observar contagem); (ii) identificar participantes inativos (por exemplo, buscando com o filtro de usuários com no máximo 2 viagens); (iii) selecionar participantes altamente engajados para entrevistas qualitativas (por exemplo, buscando com o filtro de usuários com no mínimo 30 viagens); (iv) auditar atribuições após rotação semanal (exportar CSV, comparar com planilha esperada). A Figura 3.5 ilustra esta funcionalidade (dados fictícios para fins ilustrativos).

Usuários das Coortes

Abaixo você pode visualizar e filtrar os usuários por coorte:

ID Pessoa

Nome

Email

Filtrar por ID de Pessoa

Filtrar por Nome

Filtrar por Email

ID Coorte

Min. Viagens

Máx. Viagens

Filtrar por ID de Coorte

Mín. de viagens

Máx. de viagens

ID	Nome	Email	Total de Viagens	ID Coorte	
1	Usuário fictício 1	1@gmail.com	11	2	EDITAR COORTE
2	Usuário fictício 2	2@gmail.com	0	3	EDITAR COORTE
3	Usuário fictício 3	3@gmail.com	0	1	EDITAR COORTE
4	Usuário fictício 4	4@gmail.com	0	3	EDITAR COORTE
5	Usuário fictício 5	5@gmail.com	0	2	EDITAR COORTE

Página 1

ANTERIOR

PRÓXIMO

Figura 3.5: Interface de filtragem de participantes por coorte (dados fictícios).

Para correções pontuais, administrador pode editar coorte de usuário específico clicando “Editar Coorte” na linha da tabela, alterando ID para nova coorte ou deixando vazio para remover atribuição. Para operações em massa (rotação semanal), funcionalidade de upload via CSV permite atribuições em lote: arquivo com duas colunas (CPF;ID_Coorte) separadas por ponto-e-vírgula, sem cabeçalho, codificação UTF-8. Sistema valida linha por linha antes de processar: CPF deve existir, coorte destino deve estar cadastrada, CPFs não podem duplicar. Erros específicos (“CPF 12345678901 não encontrado”, “Coorte 99 não existe”) facilitam correção e reupload. Processamento atômico garante que ou todas atribuições são aplicadas ou nenhuma, prevenindo estados parciais que comprometeriam integridade experimental.

Durante piloto, rotação semanal típica seguia workflow: economistas geravam CSV com nova atribuição usando script (randomização estratificada por perfil socioeconômico e localização geográfica); administrador fazia upload via painel (Figura 3.6); sistema atualizava os registros; notificações push/email informavam participantes sobre nova remuneração vigente; exportação CSV pós-upload confirmava aplicação correta. Este processo de rotação semanal ao longo do projeto piloto evidenciou robustez e eficiência da funcionalidade.



Upload de Atribuição de Coortes

Faça upload de um arquivo CSV para atribuir coortes a múltiplos usuários de uma só vez. O arquivo deve ter duas colunas separadas por ponto e vírgula (;) no formato: **CPF;ID_COORTE**

Exemplo de linha: 12345678901;2

CLIQUE PARA SELECIONAR O ARQUIVO CSV

ENVIAR E ATRIBUIR COORTES

Figura 3.6: Interface de upload de atribuição de coortes em lote (dados fictícios).

O botão de exportação gera CSV completo com ID, nome, CPF, email, ID da coorte, e valor de remuneração atual. O formato compatível com ferramentas estatísticas e facilita análises, o nome do arquivo gerado inclui a data da exportação, importante para criar um histórico versionado. A Figura 3.7 mostra esta funcionalidade (dados fictícios para fins ilustrativos).



Exportar Dados dos Usuários

Faça o download de um arquivo CSV contendo todos os usuários do sistema com suas informações de coorte. O arquivo incluirá: **ID, Nome, CPF, Email, ID da Coorte e Valor da Remuneração**. O delimitador do CSV é o ponto e vírgula (;).

BAIXAR CSV DE USUÁRIOS

Figura 3.7: Interface de exportação de coortes (dados fictícios).

O sistema implementa múltiplas salvaguardas: coortes não podem ser deletadas se possuem membros (previne perda accidental de atribuições); mudanças de remuneração em coorte ativa geram log detalhado (rastreabilidade de alterações durante experimento); tentativas de atribuir participante a coorte inexistente são rejeitadas (impossibilita referências órfãs); exportações registram timestamp e usuário que solicitou (auditoria de acesso a dados sensíveis).

A infraestrutura de gerenciamento de coortes no painel viabilizou análises sobre o impacto diferencial das remunerações. Os dados coletados durante o piloto permitirão investigar relações dose-resposta (comparando número de viagens entre diferentes valores de remuneração), validar a estratégia de rotação semanal, examinar retenção de participantes em cada coorte, e explorar heterogeneidade de efeitos por perfis socioeconômicos. Estas descobertas fundamentarão documento de recomendações para regulamentação da Lei Municipal 16.547/2016.

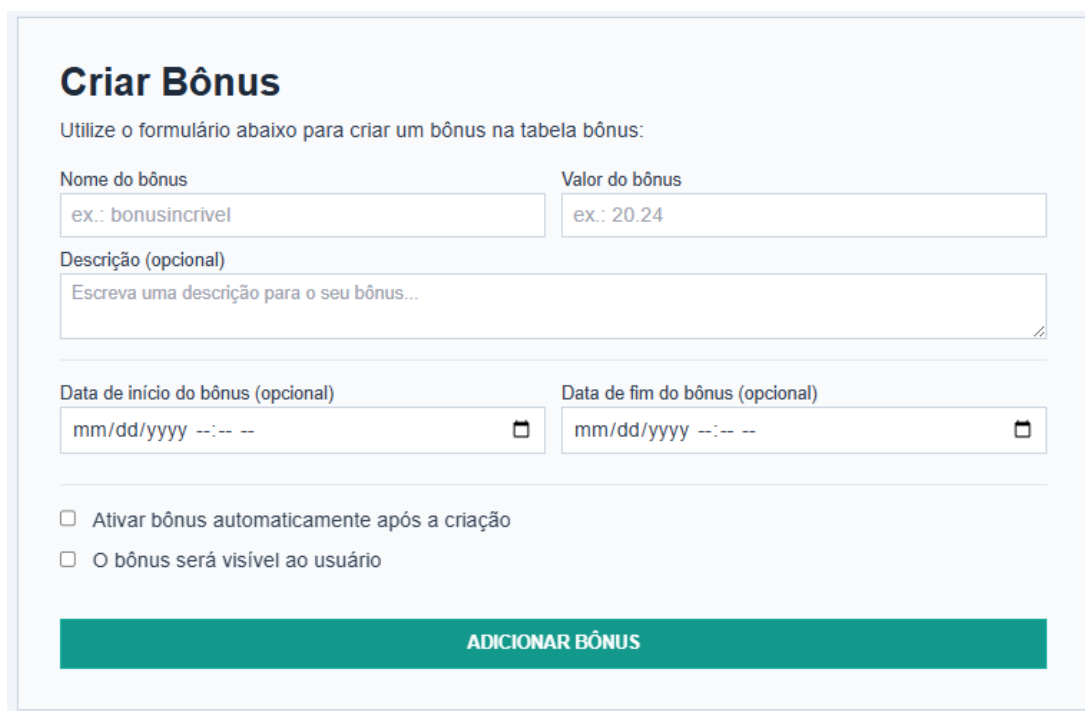
3.3.4 Bônus

No contexto do piloto BikeSP, os bônus representam uma ferramenta complementar de incentivo financeiro aos participantes, independente das remunerações por quilômetro pedalado. Conforme planejado no desenho do experimento, os bônus servem a múltiplos propósitos: (i) recompensar a conclusão do curso gratuito “Pedale com Segurança” oferecido pelo Centro de Treinamento e Educação de Trânsito (CETET) da CET, que treinou centenas de participantes do piloto em técnicas de ciclismo seguro no trânsito urbano; (ii) incentivar o preenchimento dos questionários qualitativos aplicados em três momentos do piloto, garantindo maior taxa de resposta e riqueza de dados; (iii) permitir ajustes administrativos e correções pontuais de pagamento quando necessário; e (iv) viabilizar eventuais campanhas promocionais temporárias para estimular a participação em momentos estratégicos do experimento.¹

O painel administrativo oferece três funcionalidades principais relacionadas a bônus: criação de novos tipos de bônus, atribuição de bônus a usuários, e visualização e gerenciamento dos bônus cadastrados. A arquitetura do sistema foi projetada para oferecer flexibilidade e controle sobre os incentivos.

Os administradores podem cadastrar novos tipos de bônus informando nome único (identificador), valor em reais, descrição textual, período de validade (datas de início e fim, opcionais), e dois flags de controle: se o bônus deve ser ativado automaticamente após criação, e se será visível aos usuários no aplicativo móvel. A distinção entre bônus ativos/inativos e visíveis/invisíveis oferece grande flexibilidade: bônus inativos não podem ser concedidos (útil para pausar temporariamente um incentivo), enquanto bônus invisíveis, mesmo quando concedidos e gerando remuneração, não aparecem na listagem do aplicativo do usuário (útil para ajustes administrativos discretos ou correções pontuais). A Figura 3.8 mostra o formulário de criação (dados fictícios para fins ilustrativos).

¹ Uma foto do curso “Pedale com Segurança” deveria ser incluída aqui, mostrando participantes do piloto BikeSP durante as atividades de treinamento oferecidas pelo CETET.



Criar Bônus

Utilize o formulário abaixo para criar um bônus na tabela bônus:

Nome do bônus	Valor do bônus
ex.: bonusincrível	ex.: 20.24

Descrição (opcional)

Escreva uma descrição para o seu bônus...

Data de início do bônus (opcional)	Data de fim do bônus (opcional)
mm/dd/yyyy --:-- --	mm/dd/yyyy --:-- --

☐ Ativar bônus automaticamente após a criação

☐ O bônus será visível ao usuário

ADICIONAR BÔNUS

Figura 3.8: Formulário de criação de bônus no painel administrativo (dados fictícios).

Para conceder um bônus a participantes, o administrador informa os IDs dos usuários (separados por vírgula) e seleciona o bônus desejado. O sistema valida diversas condições antes de processar a concessão: o bônus deve estar ativo, não pode ter expirado (data de término não ultrapassada), todos os usuários devem existir, e cada usuário pode receber cada bônus apenas uma vez. Ao conceder um bônus com sucesso, o sistema não apenas registra a concessão, mas também cria automaticamente uma remuneração correspondente ao valor do bônus e a insere na fila de pagamentos (para posterior transferência aos cartões Bilhete Único via Loja Virtual SPTrans). Caso algum usuário não atenda aos critérios, o sistema reporta sucesso parcial, detalhando quantos usuários receberam o bônus e quais apresentaram erros (por exemplo, “5 sucessos, 2 erros: Usuário 101 já possui este bônus; Usuário 203 não existe”). Esta funcionalidade de concessão em lote reduz significativamente o tempo necessário para operações recorrentes, como conceder o bônus do curso de segurança para todos os participantes que o concluíram em determinada semana. A Figura 3.9 ilustra esta interface (dados fictícios para fins ilustrativos).

Atribuir Bônus

Utilize o formulário abaixo para conceder um bônus previamente criado para um ou mais usuários. Insira os IDs separados por vírgula (ex: 1,2,3). Caso o bônus não exista, crie-o através da tabela anterior. (O formulário abaixo cria entradas na tabela PESSOABONUS):

IDs das Pessoas (separados por vírgula)

ex.: 15,16,17

ID do bônus

ex.: 20

Data de concessão do bônus (opcional)

mm/dd/yyyy --:-- --

ATRIBUIR BÔNUS AO USUÁRIO

Figura 3.9: *Formulário de atribuição de bônus aos participantes (dados fictícios).*

A tela de listagem exibe todos os bônus cadastrados em formato tabular, apresentando ID, nome, valor, status de ativação, visibilidade, descrição, período de validade e ações disponíveis. A interface oferece busca por nome e paginação (5 itens por página). Cada linha da tabela possui um botão para ativar ou desativar o bônus: bônus ativos exibem botão vermelho “Desativar”, enquanto bônus inativos exibem botão verde “Ativar”. Importante notar que desativar um bônus não remove concessões já realizadas nem afeta remunerações já geradas; apenas impede novas concessões daquele tipo de bônus. Para facilitar a identificação dos IDs de usuários ao atribuir bônus, a aba de bônus também oferece uma consulta integrada à tabela de usuários, com busca por nome ou ID, ordenação por qualquer coluna e paginação configurável da mesma forma que a tela de usuários. A Figura 3.10 mostra a listagem de bônus (dados fictícios para fins ilustrativos).

Bônus								
Pesquisar por nome...								
ID	Nome	Valor	Ativo?	Visível?	Descrição	Data de início	Data de término	Ações
1	Bônus inativo	R\$ 1,00	Não	Não	-	16/05/2025 11:33	16/08/2025 11:33	ATIVAR
2	Bônus de curso	R\$ 10,00	Sim	Sim	-	12/09/2025 06:36	-	DESATIVAR
3	Bônus invisível	R\$ 2,50	Sim	Não	-	12/09/2025 06:41	-	DESATIVAR
4	Bônus por questionário	R\$ 5,00	Sim	Não	-	12/09/2025 06:44	-	DESATIVAR

Página 1

ANTERIOR PRÓXIMO

Figura 3.10: Listagem de bônus cadastrados com funcionalidades de busca, paginação e ativação/desativação (dados fictícios).

O aplicativo Android dos participantes consome a API POST `/api/exibeBonus/` para listar os bônus concedidos ao usuário autenticado. A API retorna apenas bônus marcados como visíveis, garantindo que ajustes administrativos invisíveis não apareçam para os usuários finais. Esta separação entre visibilidade e concessão oferece aos gestores do piloto a flexibilidade de realizar correções financeiras sem necessariamente comunicá-las explicitamente aos participantes.

Durante a execução do piloto, a funcionalidade de bônus mostrou-se fundamental para diversos cenários operacionais: concessão em lote do bônus do curso de segurança para centenas de participantes que o concluíram simultaneamente; distribuição dos bônus dos questionários ao final de cada período de coleta de dados qualitativos; correções pontuais quando erros de geolocalização resultaram em viagens rejeitadas indevidamente (criando-se bônus invisíveis de ajuste); A flexibilidade do sistema permitiu à equipe operacional responder rapidamente a diferentes necessidades sem requerer desenvolvimento de novas funcionalidades ou intervenções manuais no banco de dados.

3.3.5 Remuneração

A Lei Municipal 16.547/2016 estabeleceu que o pagamento aos ciclistas do programa BikeSP seria realizado através de créditos de transporte público, inseridos diretamente nos cartões Bilhete Único dos participantes. Esta escolha reflete a intenção do governo municipal de integrar a bicicleta ao sistema de transporte público, permitindo que os incentivos sejam utilizados para deslocamentos multimodais (bicicleta + ônibus/metrô). O modelo de remuneração por quilômetro pedalado, adotado também em iniciativas internacionais como em Itajaí e na Holanda, visa recompensar a distância economizada ao substituir outros modos de transporte, facilitando análises de impacto em poluição,

tráfego e saúde pública.

No desenho experimental do piloto, optou-se por valores de R\$ 0,30 e R\$ 0,60 por quilômetro (equivalentes a 1/16 e 1/8 de uma passagem de transporte público, respectivamente), com limite de duas viagens remuneradas por dia. Estes valores foram baseados em estudo do Banco Mundial sobre disposição a pagar por ciclismo em São Paulo ([WORLD BANK, 2022](#)), que identificou que remunerações entre R\$ 2,00 e R\$ 3,00 por viagem aumentavam em 38,4% a probabilidade de escolher bicicleta sobre o modo atual, efeito comparável à presença de ciclovias (43,4%). A remuneração não foi diferenciada por perfil de participante durante o piloto, pois um dos objetivos do experimento era justamente compreender como o impacto da remuneração varia segundo diferentes perfis demográficos e socioeconômicos.

O sistema de remuneração implementado no painel administrativo gerencia três estados distintos do dinheiro: (i) `aguardandoEnvio` — valores de viagens aprovadas, bônus concedidos e contestações deferidas que ainda não foram enviados à SPTrans; (ii) `aguardandoResposta` — valores já enviados à SPTrans através de arquivo CSV, aguardando confirmação de creditação; e (iii) `concedido` — valores confirmados pela SPTrans como efetivamente creditados nos cartões Bilhete Único dos participantes. Esta separação em estados permite rastreabilidade completa do fluxo financeiro e reconciliação precisa entre solicitações, confirmações e valores efetivamente pagos.

A funcionalidade de geração de CSV para pagamento compila todos os valores em estado `aguardandoEnvio` de participantes com Bilhete Único ativo. O arquivo gerado segue formato específico exigido pela SPTrans: duas colunas delimitadas por ponto e vírgula (número do Bilhete Único e valor), sem cabeçalho. O nome do arquivo inclui data e hora completa (exemplo: `pagamentos05-10-2024-14-30-25.csv`) para garantir unicidade e rastreabilidade. Ao gerar o CSV, o sistema automaticamente move os valores de `aguardandoEnvio` para `aguardandoResposta`, evitando duplicação em envios subsequentes. Esta operação é atômica e registrada em log para auditoria. A Figura 3.11 mostra a interface de geração (dados fictícios para fins ilustrativos).



Figura 3.11: Interface de geração de lista de créditos a pagar via SPTrans (dados fictícios).

O arquivo CSV gerado é enviado manualmente à SPTrans. A SPTrans credita os valores nos cartões Bilhete Único dos participantes e retorna um arquivo CSV de comprovação contendo os bilhetes efetivamente creditados e os valores pagos. Este processo manual, embora não automatizado, mostrou-se adequado para a escala do piloto (cerca de 1.200 participantes) e garante supervisão humana sobre transações financeiras, reduzindo riscos de erros sistêmicos em larga escala.

Ao receber o comprovante da SPTrans, o administrador realiza upload do arquivo CSV no painel. O sistema executa validação rigorosa linha por linha: verifica formato (exatas duas colunas), existência do Bilhete Único no banco de dados, e validade numérica do valor. Qualquer erro detectado aborta imediatamente o processamento completo, impedindo confirmações parciais que poderiam gerar inconsistências. Apenas quando todas as linhas são validadas com sucesso, o sistema processa a confirmação: move valores de “aguardando resposta” para “concedido”, registra o histórico de pagamento como confirmado, e armazena a data e valor de cada operação. Mensagens específicas de erro (“Arquivo com mais de duas colunas”, “Formato incorreto de valor”, “Bilhete não encontrado”) facilitam correção rápida de problemas antes de reprocessar. A Figura 3.12 ilustra a interface de upload (dados fictícios para fins ilustrativos).

Upload de Pagamentos Realizados

O arquivo CSV dos pagamentos realizados deve ter duas colunas (sem header) com o padrão "bilhete-único";"XX,YY". Ou seja, as linhas tem um número de bilhete único, um ponto e vírgula como separador, e um valor com duas casas decimais que foi pago ao usuário.

CLIQUE PARA SELECIONAR O ARQUIVO CSV

ENVIAR E CONFIRMAR PAGAMENTOS

Figura 3.12: Interface de upload de comprovante de pagamentos realizados pela SPTrans (dados fictícios).

O sistema armazena, para cada Bilhete Único, os três estados de remuneração (concedido, aguardando envio, aguardando resposta) além da identificação do participante e status de ativação do cartão. Mantém também registro imutável de todos os envios de pagamento (identificados por número do bilhete e data), permitindo múltiplos pagamentos ao mesmo participante ao longo do tempo. Restrições do banco de dados garantem que valores sejam sempre não negativos e que transições de estado sigam apenas fluxos válidos. Toda operação financeira é registrada em log incluindo ID do administrador responsável, garantindo trilha de auditoria completa e rastreabilidade para conformidade com procedimentos do experimento.

O painel oferece visibilidade sobre valores pendentes de envio, valores aguardando confirmação da SPTrans, e total pago aos participantes. Durante o piloto, estes indicadores foram fundamentais para planejamento de fluxo de caixa e comunicação com participantes sobre prazos esperados de creditação.

O sistema contempla cenários atípicos que ocorreram durante operação: (i) pagamentos parciais, quando SPTrans credita valor inferior ao solicitado, mantendo diferença pendente de confirmação para reenvio posterior; (ii) bilhetes inválidos ou bloqueados, que não aparecem no CSV de retorno da SPTrans e requerem investigação manual com o participante; (iii) múltiplos pagamentos no mesmo dia, permitidos via timestamp distinto na chave primária; e (iv) valores zerados no CSV, que são ignorados sem gerar erro. A natureza atômica e idempotente do processamento de confirmações permite reprocessar o mesmo CSV sem duplicações, útil em casos de falhas de comunicação ou necessidade de revalidação.

3.3.6 Contestações

O algoritmo de validação automática de viagens implementado no backend Flask rejeitava viagens que não atendiam critérios predefinidos: distância < 1km (consideradas muito curtas para deslocamento cotidiano), origem ou destino fora das localizações cadastradas (potencial fraude ou uso recreacional não remunerável), ou trajetória implausível (teletransportes indicando falha de GPS). Embora estes critérios reduzissem tentativas de

fraude, geravam falsos positivos: viagem legítima rejeitada por erro temporário de GPS, localização ligeiramente deslocada devido a imprecisão de geocodificação, ou participante que iniciou pedalada próximo mas não exatamente na localização cadastrada. Sistema de contestações oferece canal para reverter rejeições injustas, balanceando a eficiência da automação com a supervisão humana.

No aplicativo móvel, participantes cujas viagens foram reprovadas visualizam botão “Contestar” na tela de detalhes da viagem. Ao clicar, preenchem campo de justificativa explicando por que consideram a viagem válida (ex: “Saí de casa normalmente, mas GPS demorou para conectar”, “Trabalho fica dentro de prédio comercial, endereço cadastrado é da entrada principal”). A contestação é enviada e cria registro com status pendente de análise, visível imediatamente na interface administrativa.

A tela administrativa de contestações exibe tabela paginada com ID da pessoa, ID da viagem, data, justificativa, e status (Pendente/Sim/Não). Busca textual cobre todos campos, útil para localizar contestações de participante específico (“busca: 123”) ou por palavra-chave na justificativa (“busca: GPS”). Ao clicar em linha, sistema destaca visualmente e carrega detalhes completos abaixo da tabela, incluindo nome do participante, justificativa integral, e mapa interativo renderizando trajeto GPS da viagem contestada. Visualização do trajeto revelou-se decisória: administrador constata se rota realmente conecta origem-destino, se passou por vias cicláveis, ou se apresenta anomalias (ex: linha reta implausível indicando falha de GPS). A visualização dos trajetos será melhor descrita na seção viagens. A Figura 3.13 mostra a interface de listagem (dados fictícios para fins ilustrativos).

Contestações

Utilize esta seção para visualizar e gerenciar as contestações pendentes às viagens. Você pode aprovar ou rejeitar cada contestação após analisar os detalhes fornecidos.

Lista de Contestações

ID Pessoa	ID Viagem	Data	Justificativa	Aprovada
1	1	15/04/25	Falha no GPS	Não
1	2	15/04/25	Iniciei um pouco distante	Pendente
1	3	15/04/25	Endereço cadastrado errado	Sim

Página 1

ANTERIOR

PRÓXIMO

Figura 3.13: Interface de listagem de contestações com busca e paginação (dados fictícios).

Após analisar mapa e justificativa, administrador pode aprovar contestação clicando botão verde com confirmação de duplo clique (prevenção contra aprovação acidental). Sistema executa validações adicionais: contestação não pode ter sido processada anteriormente, viagem deve ter origem e destino definidos e distintos, usuário deve estar em coorte ativa, e distância entre origem-destino ≥ 1 km. Cálculo de remuneração segue fórmula oficial: $\text{distancia_km} = \min(8000, \text{distancia_metros}) / 1000$; $\text{remuneracao} = \text{distancia_km} * \text{valor_por_km_da_coorte}$. Distância é limitada em 8km (viagens mais longas recebem teto de 8km) para controlar orçamento. Ao aprovar, sistema: atualiza status da viagem para “Aprovado”, salva valor de remuneração, adiciona crédito à fila de pagamentos, marca a contestação como aprovada, e opcionalmente salva resposta textual do administrador (feedback ao participante). Mensagem de sucesso confirma: “Contestação da viagem 12345 aprovada com remuneração de R\$ 4,50”. A Figura 3.14 ilustra a interface de aprovação/rejeição (dados fictícios para fins ilustrativos).

Detalhes da Contestação

ID Viagem: 2

ID Pessoa: 1

Nome: Usuário fictício 1

Data: 15/04/25

Justificativa: Iniciei um pouco distante

Aprovado: Pendente

Resposta:

Ok

APROVAR

REJEITAR

Figura 3.14: Interface de aprovação/rejeição de contestação (dados fictícios).

O administrador pode rejeitar contestação (botão vermelho, também com confirmação dupla) quando: trajeto não conecta localizações cadastradas (participante tentou viagem recreacional), distância insuficiente mesmo com tolerância, ou justificativa indica má-fé (“não sabia que precisava ativar GPS”). Sistema atualiza contestação para reprovada, salva resposta opcional explicando motivo da rejeição (ex: “Trajeto iniciou 2km distante da localização cadastrada. Por favor, inicie viagens próximo aos endereços registrados.”), e mantém viagem com status reprovado original. Campo de resposta é exibido no aplicativo móvel, permitindo participante compreender razão e ajustar comportamento futuro.

O sistema implementa múltiplas validações com mensagens específicas orientando correção: “CONTESTAÇÃO INVÁLIDA” (ID não encontrado), “CONTESTAÇÃO JÁ GERENCIADA” (processamento duplicado prevenido), “VALOR FALTANDO PARA VIAGEM COM ORIGEM OU DESTINO DESCONHECIDO” (dados incompletos na viagem), “VALOR FALTANDO PARA VIAGEM COM ORIGEM == DESTINO” (mesmo local), “PESSOA SEM GRUPO PESQUISA” (usuário não atribuído a coorte), “DISTANCIA MENOR QUE 1KM” (critério mínimo não atendido). Estas mensagens, além de prevenir estados inconsistentes no banco de dados, comunicam ao administrador qual condição falhou, permitindo diagnóstico e correção manual (ex: atribuir usuário a coorte antes de reaprovar contestação).

3.3.7 Viagens

Durante os seis meses do piloto BikeSP, foram registradas mais de 29.000 viagens pelo aplicativo Android dos participantes, totalizando mais de 150.000 km pedalados. Este volume de dados demandou interface robusta para visualização, busca e análise pela equipe de suporte e pesquisadores. A tela de viagens tornou-se ferramenta essencial para diagnóstico de problemas reportados por usuários (“minha viagem não apareceu”, “valor creditado está incorreto”), validação de contestações, e análises exploratórias para publicações científicas.

A tabela de viagens apresenta oito colunas principais: ID da viagem (identificador único), ID do usuário, data da viagem, deslocamento em metros, IDs de origem e destino (referenciando localizações cadastradas), status (aprovada/reprovada), e remuneração calculada em reais. O sistema consulta dados de viagens vinculando informações de pessoas e localizações de origem e destino, utilizando técnicas de paginação eficiente para responsividade. Ordenação padrão por ID decrescente exibe viagens mais recentes primeiro, padrão adequado para monitoramento operacional diário.

O campo de busca permite filtrar viagens por ID de usuário específico (busca exata). Esta funcionalidade atende cenário operacional frequente: participante relata problema com viagem; administrador identifica ID do usuário na tela de participantes; busca todas viagens daquele usuário; localiza a viagem em questão (geralmente reconhecível por data/horário ou trajeto). A Figura 3.15 mostra a interface de listagem (dados fictícios para fins ilustrativos).

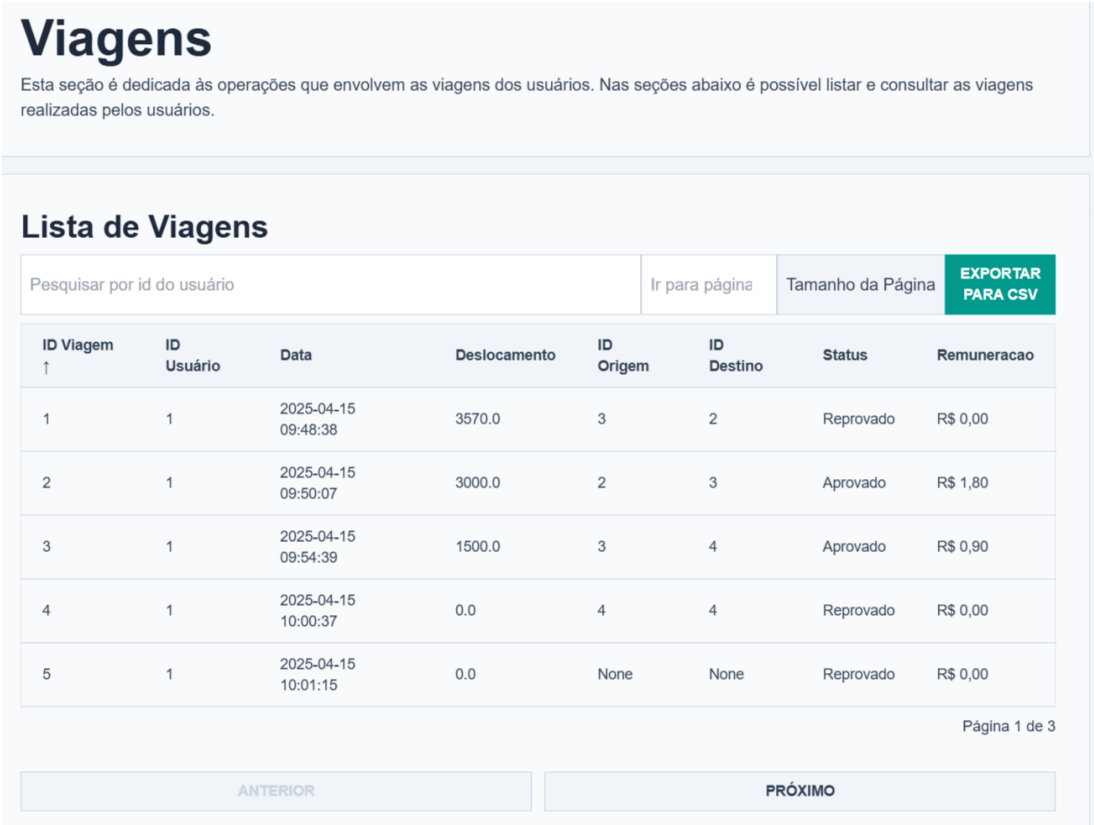


Figura 3.15: Interface de listagem de viagens com funcionalidades de busca, ordenação e paginação (dados fictícios).

Ao clicar em qualquer linha da tabela, a interface renderiza mapa interativo (biblioteca Leaflet.js com tiles OpenStreetMap) exibindo trajeto completo da viagem abaixo da tabela. O mapa plota marcadores para cada ponto GPS coletado pelo aplicativo durante a viagem, conectados por linha azul na ordem temporal. Popups nos marcadores exibem timestamp de cada ponto. Zoom automático enquadra toda rota, e controles padrão (zoom, pan, scroll) permitem inspeção detalhada. Esta funcionalidade revelou-se crítica para validação de

viagens contestadas: administradores visualizavam se trajeto realmente conectava origem e destino declarados, se passava por vias adequadas para ciclismo, e se não apresentava anomalias (ex: teletransportes indicando GPS incorreto). A Figura 3.16 ilustra a visualização de um trajeto (dados fictícios para fins ilustrativos).

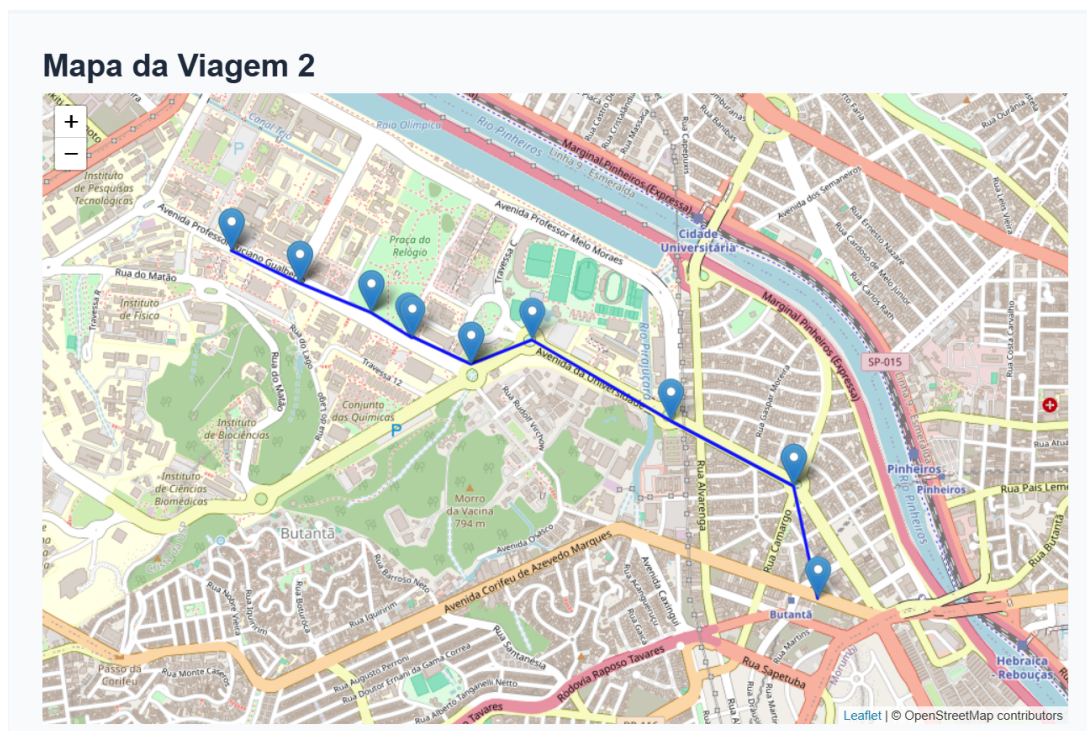


Figura 3.16: Visualização de trajeto de viagem no mapa interativo com marcadores de pontos GPS (dados fictícios).

O botão de exportação gera arquivo CSV contendo todas as informações exibidas (respeitando filtros de busca ativos) para análise em ferramentas externas. O formato inclui cabeçalho descritivo e delimitador ponto-e-vírgula. Durante piloto, esta funcionalidade foi utilizada semanalmente por economistas do projeto para análises estatísticas: modelos de regressão investigando impacto das coortes no número de viagens, análises de série temporal identificando padrões de uso ao longo da semana, e visualizações geoespaciais em GIS identificando origens/destinos mais frequentes. A possibilidade de exportar dados filtrados (ex: apenas viagens de determinado usuário ou período) reduziu necessidade de consultas diretas ao banco de dados para análises.

A ordenação por diferentes colunas viabiliza análises rápidas: ordenar por deslocamento descendente identifica viagens mais longas (útil para identificar outliers ou comportamento atípico); ordenar por data permite inspeção cronológica (verificar se sistema registrou viagens durante manutenções); ordenar por status agrupa aprovadas vs. reprovadas (calcular taxa de aprovação manual); ordenar por remuneração identifica viagens de maior valor. Durante análise dos resultados do piloto, observou-se que 9 das 10 viagens mais longas (identificadas via ordenação por deslocamento) ocorreram em finais de semana sem remuneração, evidência importante de retenção intrínseca do hábito de pedalar para além do incentivo financeiro.

A configuração padrão de 10 viagens por página balanceia densidade de informação e tempo de carregamento. Com 29.000+ registros, carregar todas viagens simultaneamente seria inviável; paginação server-side (LIMIT/OFFSET no PostgreSQL) mantém responsividade independente do volume total. Indicador “Página X de Y” fornece senso de escala, reforçando necessidade de busca e ordenação para localizar informação relevante. Opções de 5, 15 ou 20 itens por página atendem diferentes casos de uso: densidade baixa para inspeção cuidadosa com múltiplos mapas abertos; densidade alta para varredura rápida de padrões.

Cenário 1: Diagnóstico de viagem não creditada — participante relata viagem realizada ontem não apareceu no extrato; administrador busca por ID do usuário, filtra viagens recentes (ordenação por data), identifica viagem com status “reprovada”; visualiza mapa, constata que trajeto passou por localizações não cadastradas; orienta participante a atualizar endereços. *Cenário 2: Validação de contestação* — usuário contesta rejeição de viagem; administrador abre viagem contestada, visualiza mapa, confirma que trajeto conecta origem e destino cadastrados; aprova contestação, gerando remuneração retroativa.

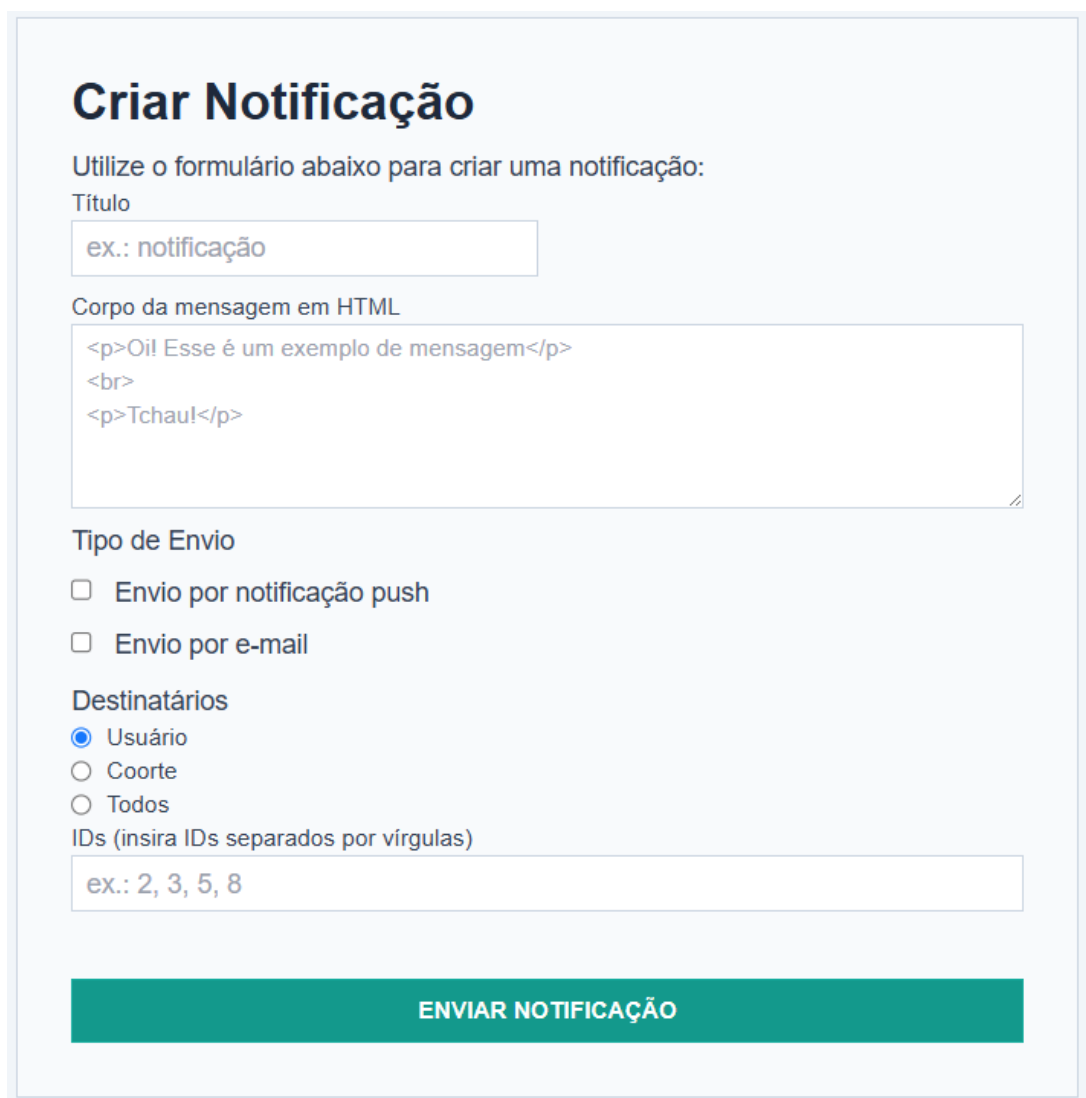
3.3.8 Notificações

Com 1.217 participantes distribuídos geograficamente por São Paulo e engajados em experimento com rotação semanal de coortes, estabelecer canal eficaz de comunicação mostrou-se crítico. Eventos requerendo notificação incluíam: início de nova coorte (informar mudança de remuneração), disponibilidade de pagamento (créditos depositados no Bilhete Único), manutenções programadas no sistema, lembretes de preenchimento de questionários qualitativos, e esclarecimentos sobre regras do experimento após dúvidas recorrentes. Sistema de notificações implementado no painel oferece dois canais simultâneos: push notifications via Firebase Cloud Messaging (FCM) para aplicativo Android, e email via SMTP (Gmail) para endereços cadastrados.

A interface oferece três estratégias de direcionamento: (i) envio individual para IDs específicos separados por vírgula (ex: comunicação de suporte técnico, “participantes 123, 456 e 789: problema reportado foi corrigido”); (ii) envio por coorte via IDs de grupos experimentais (ex: “participantes da coorte 2: esta semana sua remuneração é R\$ 0,60/km”); e (iii) broadcast para todos participantes cadastrados (ex: “sistema em manutenção dia 15 das 02h-04h”). Esta granularidade atende necessidades distintas: comunicação individualizada reduz ruído para participantes não afetados; segmentação por coorte mantém integridade do desenho experimental (evitando confusão sobre qual valor de remuneração aplica-se a cada grupo); broadcast garante que informações críticas alcancem todos.

O formulário de criação requer título (usado como assunto de email e cabeçalho de push notification) e corpo da mensagem. Campo de corpo aceita HTML completo, permitindo formatação rica em emails: parágrafos (<p>), quebras de linha (
), negrito (), listas, e links. Suporte a HTML mostrou-se valioso para comunicações longas ou complexas, como FAQ respondendo dúvidas recorrentes ou instruções passo-a-passo para resolver problemas comuns. Push notifications, devido a limitações de espaço em dispositivos móveis, exibem apenas texto plano, adequado para mensagens concisas. Checkbox independentes para cada canal (push e/ou email) permitem escolher meio mais apropriado: push para alertas urgentes que requerem atenção imediata; email para informações detalhadas que

usuário pode consultar posteriormente. A Figura 3.17 mostra a interface de criação (dados fictícios para fins ilustrativos).



O formulário, intitulado "Criar Notificação", contém os seguintes campos e opções:

- Título:** Campo de texto com o exemplo "ex.: notificação".
- Corpo da mensagem em HTML:** Área de texto com o exemplo: `<p>Oi! Esse é um exemplo de mensagem</p>
<p>Tchau!</p>`.
- Tipo de Envio:** Duas opções de caixa de seleção: ☐ "Envio por notificação push" e ☐ "Envio por e-mail".
- Destinatários:** Três opções de radio button: ☒ "Usuário", ☐ "Coorte" e ☐ "Todos".
- IDs (insira IDs separados por vírgulas):** Campo de texto com o exemplo "ex.: 2, 3, 5, 8".
- Botão:** Um botão verde com o texto "ENVIAR NOTIFICAÇÃO".

Figura 3.17: Formulário de criação e envio de notificações para participantes (dados fictícios).

Para push notifications, o sistema consulta tabela de tokens FCM associados a cada participante (registrados durante primeira abertura do aplicativo), envia payload via API do Firebase, e notificação aparece na bandeja de notificações Android do participante. Falhas ocasionais ocorrem quando aplicativo é desinstalado ou token expira. Para emails, sistema busca endereços cadastrados, envia via protocolo SMTP autenticado em conta Gmail institucional, posicionando destinatários em BCC (cópia oculta para privacidade) e destinatário padrão configurável (tipicamente `bikesp-app@ime.usp.br`) no campo "Para". Esta arquitetura preserva privacidade (nenhum participante vê emails de outros) e profissionalismo (remetente institucional aumenta confiabilidade, reduzindo classificação como spam).

A funcionalidade de cópia oculta para email do administrador que enviou a notificação

(mencionada em documentação como “enviadas em cópia oculta para o email do administrador”) cria trilha de auditoria e permite verificação imediata de conteúdo e destinatários. Este registro mostrou-se útil para resolver disputas e manter histórico de comunicações para análise de engajamento.

O sistema valida que ao menos um canal (push ou email) está selecionado; que IDs de usuários/coortes são numéricos; e que campo de IDs é obrigatório apenas quando estratégia “Usuário” ou “Coorte” está selecionada (hidden quando “Todos” marcado). Log de todas operações registra quem enviou, para quem, quando, e conteúdo da mensagem.

O sistema de notificações foi utilizado para diversos fins: broadcasts (anúncios gerais, disponibilidade de pagamentos), notificações por coorte (lembretes de remuneração ativa, instruções específicas), e mensagens individuais (suporte técnico, correções de problemas).

3.3.9 Localizações

O desenho do experimento BikeSP especificou que apenas viagens entre localizações pré-cadastradas seriam remuneradas, visando incentivar deslocamentos cotidianos (casa-trabalho, casa-escola) em detrimento de viagens recreacionais irregulares. Durante inscrição via formulário LimeSurvey, participantes forneciam endereços textuais de até três localizações principais (tipicamente residência, trabalho/estudo, e uma terceira opcional). O sistema realizava geocodificação automática destes endereços — conversão de texto (“Av. Paulista, 1000, Bela Vista, CEP 01310-100”) para coordenadas GPS (latitude/longitude) utilizando APIs externas de geolocalização.

Para prevenir fraudes (participante alterando localizações para simular viagens longas fictícias), implementou-se regra de uma única alteração por localização após cadastro inicial. Participante que cadastrasse endereço incorreto ou mudasse de residência/trabalho poderia corrigi-lo uma vez via aplicativo; segunda tentativa de alteração seria bloqueada. Esta restrição, embora necessária para integridade experimental, gerou demandas de suporte quando usuários cometiam erros tipográficos ou mudavam endereço legitimamente após já terem usado sua cota de alteração.

A interface administrativa de localizações permite buscar por ID de participante e visualizar todas localizações cadastradas (ID, endereço completo, coordenadas, tipo). Funcionalidade crítica é reset da permissão de alteração: administrador, após validar legitimidade da solicitação via email ou telefone, pode remover o registro que controla alterações de localização, concedendo ao participante nova oportunidade de modificar endereço no aplicativo. Mensagens de feedback (“O usuário agora pode trocar uma localização” vs. “Nenhuma alteração de localização foi encontrada”) informam sucesso ou ausência de restrição ativa. Este procedimento foi utilizado durante o piloto, geralmente após mudanças residenciais ou correções de erros de digitação durante inscrição inicial.

Durante o script de inserção em lote de participantes do LimeSurvey para PostgreSQL, alguns endereços falhavam em geocodificação automática. Causas possíveis incluíam: formatação inconsistente (“Rua X, 100” vs. “Rua X|100|...”), abreviações não reconhecidas, bairros incorretos, CEPs inválidos, endereços recém-criados ausentes em bases cartográficas, e timeouts de APIs. Localizações que falhavam geocodificação eram marcadas como “inválidas” e ficavam inacessíveis no aplicativo móvel — participante visualizava

localização na lista mas não podia utilizá-la como origem/destino de viagens, bloqueando efetivamente sua participação.

Este problema revelou-se crítico: correção manual urgente foi necessária durante o piloto. A funcionalidade de correção de geolocalização tornou-se operação essencial.

A tela de localizações inválidas lista endereços que falharam, exibindo ID, ID do participante, endereço textual, tipo e apelido da localização. Funcionalidades de busca, ordenação e paginação seguem padrão estabelecido em outras telas. Para corrigir, administrador edita endereço diretamente seguindo dois formatos possíveis: (i) endereço estruturado separado por pipe (ex: Av. Paulista|1000|sala 5|Bela Vista|01310-100|São Paulo|SP), submetido novamente às APIs de geocodificação; ou (ii) coordenadas GPS diretas (ex: -23.5997136,-46.4677964), sem precisar passar pela geocodificação quando endereço é intratável por APIs mas localização é conhecida via Google Maps. Validação bem-sucedida remove localização da lista de inválidas, tornando-a imediatamente disponível no aplicativo do participante.

Workflow operacional típico incluía: cópia do endereço para Google Maps, verificação de formatação e bairro correto, ajuste se necessário, validação no sistema. Problemas recorrentes e soluções documentadas: CEPs com dígitos transpostos (buscar CEP correto online); bairros coloquiais vs. oficiais (Pinheiros vs. Alto de Pinheiros); abreviações (“Av.” expandir para “Avenida”); endereços inexistentes (usar coordenadas GPS obtidas manualmente). A disponibilidade de múltiplas APIs de geolocalização aumentava taxa de sucesso, mas timeouts ocasionais justificavam opção de inserção direta de coordenadas.

A correção de localizações inválidas foi realizada pela equipe de suporte. Diferentes abordagens foram utilizadas: reformatação com nova geocodificação automática, inserção manual de coordenadas diretas obtidas via Google Maps, ou contato com participante para clarificação de endereço.

Para iterações futuras do programa, recomendações incluem: validação de formato durante preenchimento do formulário LimeSurvey, geocodificação imediata após submissão do formulário com feedback visual ao candidato, e assistente de endereçamento integrado (autocompletar baseado em API de CEPs). A natureza manual e laboriosa da correção em lote demonstrou ser gargalo operacional não escalável: expansão para 10.000 participantes (meta de implementação municipal completa) seria inviável sem automação adicional nesta etapa.

Após a descrição detalhada da implementação do painel administrativo e suas funcionalidades, o próximo capítulo apresenta os resultados alcançados durante a execução do piloto, incluindo dados quantitativos de uso e uma avaliação do impacto do sistema nas operações.

Capítulo 4

Resultados e avaliação

Este capítulo apresenta os resultados alcançados durante a execução do piloto BikeSP. A Seção 4.1 descreve os números e conquistas do experimento, incluindo volume de viagens, participação e descobertas relevantes. A Seção 4.2 apresenta uma análise de como o painel administrativo apoiou as perguntas de pesquisa e a execução do piloto.

4.1 Resultados atingidos

O piloto alcançou resultados expressivos em termos de coleta de dados e engajamento dos participantes. Durante o pré-teste e o piloto, houve **coleta intensiva de dados** via aplicativo e painel ao longo de cerca de **dois meses** (julho e agosto). Os números do experimento foram:

- **3 mil inscrições** recebidas no processo seletivo;
- **1217 participantes** selecionados para o experimento;
- **Mais de 29 mil viagens** registradas pelo aplicativo durante o experimento;
- **Mais de 150 mil quilômetros** pedalados no total.

Um resultado particularmente relevante foi a **retenção no uso do aplicativo mesmo sem remuneração**. Entre as 10 viagens mais longas registradas, 9 foram realizadas fora dos períodos de remuneração (finais de semana), indicando engajamento genuíno dos participantes com o sistema e com a prática do ciclismo urbano, independentemente do incentivo financeiro imediato.

O projeto foi considerado um sucesso operacional. Em julho, o painel foi usado para acompanhar o uso e prestar atendimento a usuários com dificuldades; em agosto, executou-se o piloto principal em quatro semanas. A partir do segundo teste com cerca de 40 participantes, foram mapeadas novas funcionalidades críticas. Entre elas, destaca-se a **página do painel dedicada a corrigir erros de geolocalização**, cuja operação reduziu consideravelmente as reclamações dos usuários, permitindo que a equipe de suporte resolvesse problemas técnicos de forma ágil e eficiente.

Entre os ganhos observados com o uso do painel:

- **Eficiência:** atribuição de bônus em lote e ações em linha reduziram etapas repetitivas e tempo operacional.
- **Qualidade da informação:** padronização de datas/estados e visões dedicadas (Pessoas, Coortes) elevaram a confiabilidade das consultas e relatórios.
- **Governança:** listas de pagamento com trilhas de decisão e histórico de contestações facilitaram auditoria e reconciliação.
- **Suporte ao usuário:** ferramentas de correção de dados (especialmente geolocalização) reduziram reclamações e melhoraram a experiência dos participantes.

4.2 Análise

Como o painel apoia as perguntas de pesquisa e a execução do piloto (INCT INTERSCITY, 2025).

O próximo capítulo conclui este trabalho, destacando as principais contribuições e discutindo possibilidades de trabalhos futuros e expansão do sistema.

Capítulo 5

Conclusão

Este trabalho apresentou o contexto do programa Bike SP e a implementação de um painel administrativo para apoiar o piloto. As principais contribuições foram a estruturação de módulos voltados à operação e análise de dados, alinhados ao desenho experimental documentado (LIMA *et al.*, 2023).

O sistema desenvolvido permitiu a coleta bem-sucedida de dados de mais de 29 mil viagens e o gerenciamento operacional de 1217 participantes ao longo de quatro semanas de experimento. Os dados coletados através do software desenvolvido serão essenciais para a elaboração de um **documento de recomendações para a regulamentação da Lei Municipal 16.547/2016** em São Paulo, fornecendo evidências empíricas sobre a efetividade de diferentes níveis de incentivo financeiro na promoção do ciclismo urbano.

Os resultados positivos do piloto abrem possibilidades para trabalhos futuros, incluindo: (i) adaptação do sistema para suportar um número maior de participantes na implantação em larga escala do programa em São Paulo; (ii) replicabilidade do sistema em outras cidades — o sistema está sendo considerado para adoção pela Prefeitura de Fortaleza, o que demandará adaptações e generalizações do código; (iii) estudos aprofundados sobre os padrões de mobilidade identificados, perfis de usuários e efetividade comparativa dos diferentes níveis de incentivo; e (iv) refinamento contínuo das funcionalidades do painel com base no feedback da equipe operacional e dos pesquisadores.

A experiência adquirida no desenvolvimento e operação deste painel demonstra o papel fundamental de ferramentas administrativas bem projetadas na viabilização de experimentos científicos de políticas públicas de mobilidade urbana.

Referências

- [INCT INTERSCITY 2025] INCT INTERSCITY. *Projeto Piloto — BikeSP*. Comunicado e informações do piloto do BikeSP. 2025. URL: <https://intercity.org/bikesp/piloto/> (acesso em 31/08/2025) (citado nas pgs. 3, 34).
- [JEEKEL 2017] Hans JEEKEL. “Social sustainability and smart mobility: exploring the relationship”. *Transportation Research Procedia* 25 (2017). Sustentabilidade social e mobilidade inteligente, pp. 4296–4310. DOI: [10.1016/j.trpro.2017.05.254](https://doi.org/10.1016/j.trpro.2017.05.254) (citado nas pgs. 1, 4).
- [LIMA *et al.* 2023] Ana Yoon Faria de LIMA *et al.* “Cycling promotion using financial incentives: a case study in são paulo, brazil”. In: *CRBAM23 Book of Abstracts*. Design do piloto do BikeSP. 2023. URL: https://intercity.org/assets/CRBAM23_BookOfAbstracts_pgs_71_to_74-Ana-Yoon-Faria-de-Lima.pdf (acesso em 31/08/2025) (citado nas pgs. 1, 3, 35).
- [METRÔ SÃO PAULO 2017] METRÔ SÃO PAULO. *Pesquisa origem e destino 2017*. Pesquisa OD 2017. 2017. URL: <https://www.metro.sp.gov.br/pesquisa-od/> (citado na pg. 1).
- [PEARSON *et al.* 2020] Nicole PEARSON *et al.* “Guidance for conducting feasibility and pilot studies for implementation trials”. *Pilot and Feasibility Studies* 6.1 (2020). Orientações para estudos piloto, p. 167. DOI: [10.1186/s40814-020-00634-w](https://doi.org/10.1186/s40814-020-00634-w) (citado nas pgs. 1, 3).
- [PREFEITURA DE SÃO PAULO 2016] PREFEITURA DE SÃO PAULO. *Lei nº 16.547 — institui o programa bike sp no âmbito do município de são paulo*. Lei municipal que institui o programa Bike SP. 2016. URL: <https://legislacao.prefeitura.sp.gov.br/leis/lei-16547-de-21-de-setembro-de-2016> (citado na pg. 1).
- [PREFEITURA DE SÃO PAULO 2022] PREFEITURA DE SÃO PAULO. *Climate Action Plan São Paulo*. Plano de Ação Climática de São Paulo. 2022. URL: <http://www.cetsp.com.br/consultas/bicicleta/mapa-de-infraestrutura-ciclovitaria.aspx> (citado na pg. 1).
- [RUNDLE-THIELE 2009] Sharyn RUNDLE-THIELE. “Bridging the gap between claimed and actual behaviour: the role of observational research”. *Qualitative Market Research: An International Journal* 12.3 (2009). Lacuna entre comportamento declarado e real, pp. 295–306 (citado na pg. 3).

- [SHELLER 2018] Mimi SHELLER. *Mobility Justice: The Politics of Movement in An Age of Extremes*. Teoria sobre justiça de mobilidade. Verso Books, 2018. ISBN: 978-1-78873-093-8 (citado na pg. 1).
- [TENNANT 2022] Callum TENNANT. *The world's cycling nation: How the Netherlands redesigned itself as a country fit for bikes*. Sistema de incentivos na Holanda. 2022. URL: <https://www.euronews.com/next/2022/09/17/the-worlds-cycling-nation-how-the-netherlands-redesigned-itself-as-a-country-fit-for-bikes> (citado na pg. 1).
- [THE BRUSSELS TIMES WITH BELGA 2022] THE BRUSSELS TIMES WITH BELGA. *More and more people are cycling to work in Belgium*. Incentivos na Bélgica. 2022. URL: <https://www.brusselstimes.com/290497/more-and-more-people-are-cycling-to-work-in-belgium> (citado na pg. 1).
- [VANOUTRIVE e COOPER 2020] Thomas VANOUTRIVE e Erin COOPER. “How just is transportation justice theory? the issues of paternalism and production: a rejoinder”. *Transportation Research Part A: Policy and Practice* 133 (2020). Teoria da justiça de transporte, pp. 387–390 (citado na pg. 4).
- [WORLD BANK 2022] WORLD BANK. *Relatório apoio ao planejamento ciclovitário*. Estudo de preferência declarada sobre incentivos financeiros para ciclismo em São Paulo. Programa Smart Mobility São Paulo, 2022. URL: https://drive.prefeitura.sp.gov.br/cidade/secretarias/upload/transportes/Relatorio_apoio_ao_Planejamento_Ciclovitario.pdf (citado na pg. 21).

Índice remissivo

C

Captions, *veja* Legendas

Código-fonte, *veja* Floats

E

Equações, *veja* Modo matemático

F

Figuras, *veja* Floats

Floats

Algoritmo, *veja* Floats, ordem

Fórmulas, *veja* Modo matemático

I

Inglês, *veja* Língua estrangeira

P

Palavras estrangeiras, *veja* Língua es-

trangeira

R

Rodapé, notas, *veja* Notas de rodapé

S

Subcaptions, *veja* Subfiguras

Sublegendas, *veja* Subfiguras

T

Tabelas, *veja* Floats

V

Versão corrigida, *veja* Tese/Dissertação,
versões

Versão original, *veja* Tese/Dissertação,
versões