Pet Adoption Center Web Application Documentation

Тимски проект

Универзитет „Св. Кирил и Методиј" во Скопје
ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО
Supported by A1

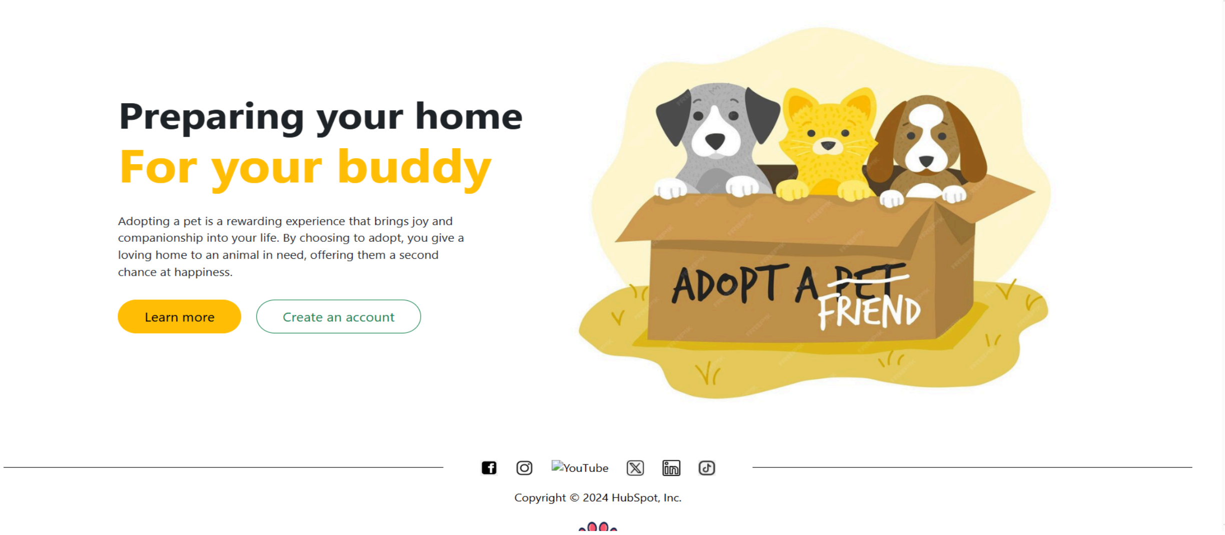
PET ADOPTION CENTER
ADOPT A FRIEND

Изработиле:
Самоил Јаќимоски (211036)
Бранко Георгиев (213077)

Ментор:
Проф. Д-р Петре Ламески

21.12.2024

Table of Contents



Preparing your home

For your buddy

Adopting a pet is a rewarding experience that brings joy and companionship into your life. By choosing to adopt, you give a loving home to an animal in need, offering them a second chance at happiness.

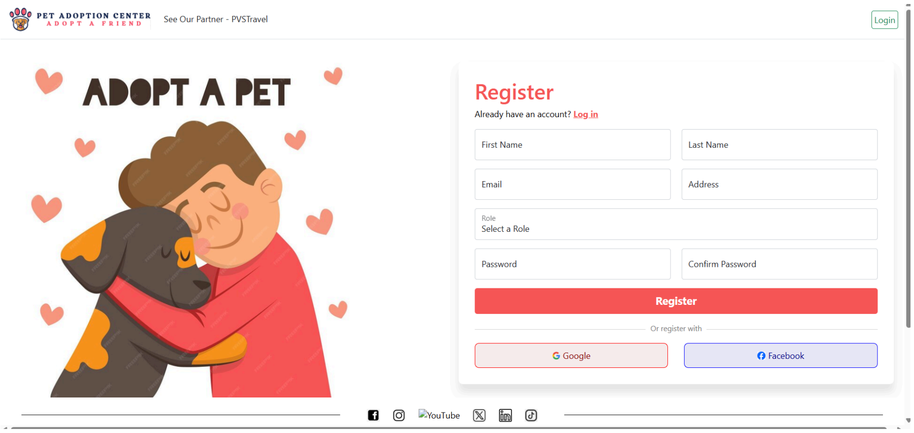
Copyright © 2024 HubSpot, Inc.

System Overview

The Pet Adoption Center web application facilitates the connection between animal shelters and potential pet adopters. It streamlines the entire adoption process from listing pets to completing payments and finalizing adoptions. The system comprises three main interfaces: one for adopters, one for shelters, and an administrative dashboard.

Core Purpose

To create a digital marketplace that simplifies pet adoption, improves visibility for shelters, provides a better experience for adopters, and ultimately helps more animals find loving homes.



Register

Already have an account? Log in

First Name

Last Name

Email

Address

Role
Select a Role

Password

Confirm Password

User Roles

Shelter

Users who represent animal shelters, rescue organizations, or other entities that have animals available for adoption.

Permissions:

- Create and manage pet listings
- Upload photos and information about available pets
- Review adoption applications
- Approve or reject applicants
- Track adoption status and manage inventory
- View payment confirmations

Adopter

Individuals seeking to adopt a pet.

Permissions:

- Browse available pets
- Filter and search for specific pets
- Submit adoption applications
- Respond to follow-up questions
- Make adoption fee payments via Stripe
- View status of their applications

Administrator

System administrators who oversee the entire platform.

Permissions:

- Access to all system functions
- User management (both shelters and adopters)
- Content moderation
- Data export capabilities
- Analytics and reporting
- System configuration

Key Features

For Shelters

- **Pet Listing Management**: Create, edit, and delete pet listings with detailed information
- **Application Review**: Evaluate adoption applications through a structured interface
- **Applicant Selection**: Choose the best-suited adopter for each pet
- **Status Tracking**: Monitor the adoption process from listing to payment completion
- **Dashboard**: Visualize shelter performance metrics and adoption statistics

For Adopters

- **Pet Discovery**: Browse through available pets with filtering options
- **Application Submission**: Complete and submit adoption questionnaires
- **Status Tracking**: Monitor application progress
- **Secure Payments**: Process adoption fees through Stripe integration
- **Favorite Pets**: Save pets of interest for later review

For Administrators

- **User Management**: Create, modify, and suspend user accounts
- **Content Oversight**: Review and moderate pet listings and user interactions
- **Data Export**: Generate and download reports in various formats
- **System Monitoring**: Track platform usage, performance, and potential issues
- **Global Configuration**: Set system-wide parameters and settings

Application Workflow

Pet Listing Process

1. Shelter creates account (if new) or logs in
2. Shelter completes pet profile with details:
    - Species, breed, age, gender
    - Health information, behavioral traits
    - Story/background
    - Photos (multiple allowed)
    - Adoption fee
3. Shelter publishes listing, making it visible to adopters

Adoption Process

1. Adopter browses available pets
2. Adopter submits application for desired pet

3. Application status changes to "Pending Review"
4. Shelter reviews all applications for the pet
5. Shelter selects preferred adopter
6. Selected adopter receives notification
7. Application status changes to "Waiting for Payment"
8. Adopter completes payment through Stripe
9. Application status changes to "Adopted"
10. Pet listing is marked as "Adopted" and removed from available pets

Administrative Workflow

1. Admin monitors system activity through dashboard
2. Reviews user reports or flagged content
3. Provides support to shelters and adopters as needed
4. Generates reports and exports data for analysis
5. Manages system configurations and updates

Technical Architecture

Backend

- **Server**: .Net 8.0 Core
- **API Architecture**: MVC (Model - View - Controller)
- **Authentication**: .Net Identity
- **Hosting Platform:** AWS Elastic Beanstalk

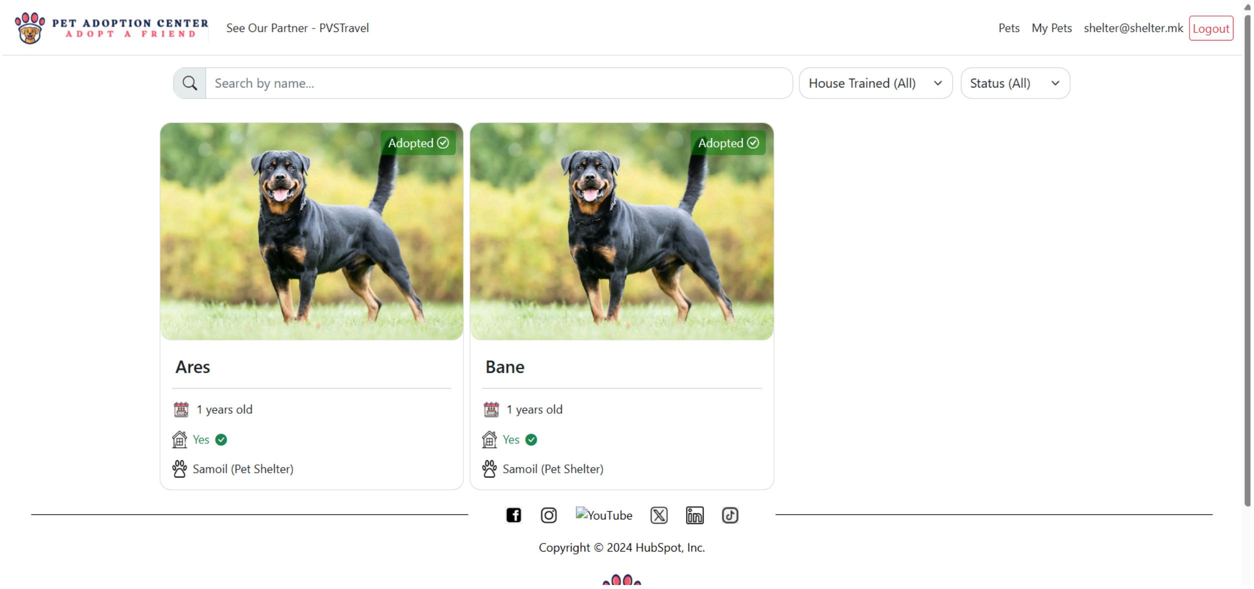
Database

- **Primary Database**: Microsoft SQL Server
- **File Storage**: Local Storage

Third-Party Integrations

- **Payment Processing**: Stripe API

User Interfaces



Adopted ✓

Ares

1 years old

Yes ✓

Samoil (Pet Shelter)

Adopted ✓

Bane

1 years old

Yes ✓

Samoil (Pet Shelter)

Copyright © 2024 HubSpot, Inc.

Shelter Interface

- **Dashboard**
  - Overview of listings, applications, and adoption metrics
  - Quick access to pending applications requiring review
  - Notifications for new applications and system updates
- **Pet Management**
  - Form-based interface for creating/editing pet listings
  - Image upload capability with preview
  - Status toggles for each pet listing
- **Application Review**
  - Split-view layout showing pet details and applicant information
- **Adoption Tracking**
  - Payment confirmation indicators
  - Archive of completed adoptions

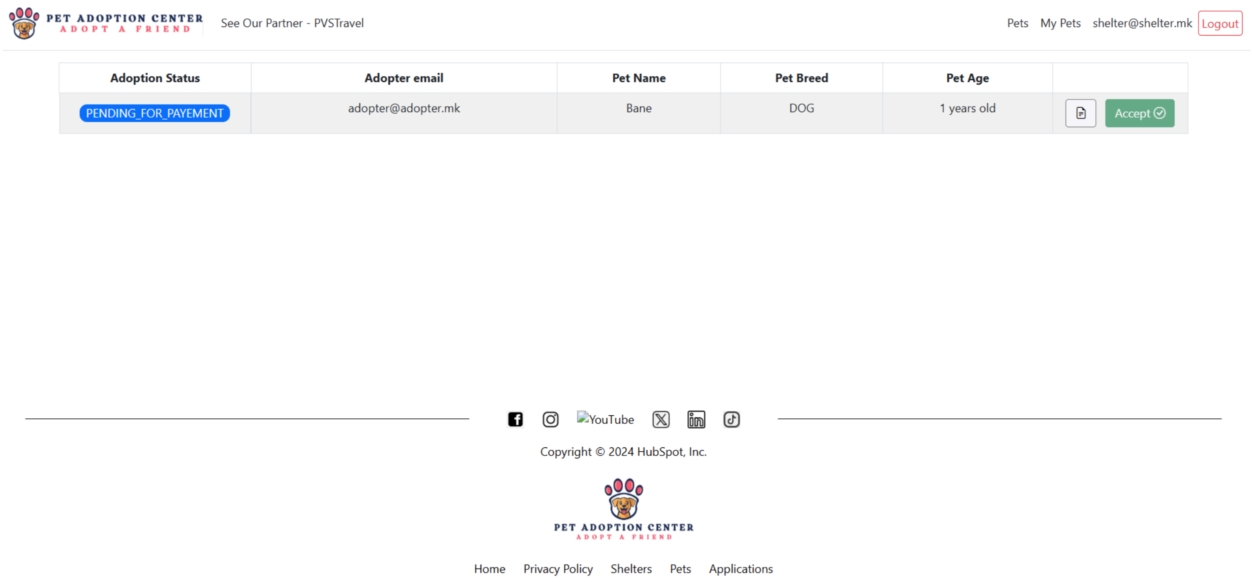
Adopter Interface

- **Pet Discovery**
  - Grid/list view of available pets with filtering options
  - Detailed pet profile pages with image galleries
- **Application**

    - Multi-step questionnaire interface
    - Application status tracking
    - Communication portal with shelters
- **Payment**
    - Secure Stripe payment form
    - Payment confirmation page

Administrative Interface

- **Dashboard / Data Export**
    - All available pets
    - Report configuration interface
    - Format (Excel)



Adoption Status	Adopter email	Pet Name	Pet Breed	Pet Age	
PENDING_FOR_PAYEMENT	adopter@adopter.mk	Bane	DOG	1 years old	Accept

API Endpoints

Pets

- `GET /pets` - List all available pets
- `GET /pets/details/:id` - Get specific pet details
- `POST /pets/create` - Create new pet listing (shelter only)
- `PUT /pets/edit/:id` - Update pet listing (shelter only)
- `DELETE /pets/delete/:id` - Remove pet listing (shelter only)

Applications

- `GET /adoptionapplications` - List applications (filtered by user role)
- `GET /adoptionapplications/details/:id` - Get specific application details
- `POST /adoptionapplications/create` - Submit new application (adopter only)
- `PUT /adoptionapplications/edit/:id` - Update application status (shelter only)

Payments

- `POST /pay?stripeEmail?stripeToken?id` - Pay order (Stripe)

Admin

- `GET /api/pacadmincontroller/getpetdetails/:id` - Get specific pet details
- `GET /api/pacadmincontroller/gemypets/:id` - List all pets for specific shelter
- `GET /api/pacadmincontroller/listallpets` - List all pets
- `PUT /api/pacadmincontroller/system/config` - Update system configuration

Database Schema

Users Table

- `Id`: Unique identifier
- FirstName: First Name
- LastName: Last Name
- Address: Full Address
- `Email`: Email address (unique)
- `Password`: Hashed password
- `PhoneNumber`: Contact number
- `Role`: Enum (adopter, shelter, admin)

Asp Net Roles

- Id: Unique identifier
- Name: Name of the role

Asp Net User Roles

- UserId: Foreign key to user
- RoleId: Foreign key to asp net role

Pets Table

- `Id`: Unique identifier
- `Name`: Pet name
- `Age`: Age in years/months
- `Species`: Species (dog, cat, etc.)
- `Breed`: Breed or mix description
- Sex: Male or Female
- IsHouseTrained: Is the pet house trained
- FavouriteThings: List of favorite things
- HomeRequirements: Home requirements for adoption
- PhotoUrl: Photo of the pet
- PetStatus: Status (available for adoption, adopted)
- `description`: Detailed description
- `Shelter_id`: Foreign key to Shelters
- `Price:` Price

AdoptionApplications Table

- `id`: Unique identifier
- `pet_id`: Foreign key to Pets
- `adopter_id`: Foreign key to Adopters
- `adoption_application_status`: Status of the application (pending, approved, rejected, waiting_for_payment, completed)
- `Question1 - Question10`: Questions with application answers

Payment Integration

Stripe Implementation

- **Client-Side**: Stripe Elements for secure card collection
- **Server-Side**: Payment Intents API for transaction processing

Payment Flow

1. Application approved by shelter
2. System creates Stripe Payment Intent
3. Adopter redirected to secure payment form
4. Payment submitted and processed by Stripe
5. Webhook notification confirms payment
6. System updates application status
7. Both parties receive confirmation

Security Considerations

Authentication & Authorization

- Role-based access control (RBAC)
- Password strength requirements
- Password hashing with bcrypt/Argon2
- Session timeout and refresh token mechanism

Data Protection

- HTTPS for all communications
- Database encryption for sensitive fields
- Input validation and sanitization

Administrative Functions

Data Management

- Export data (Excel)

Deployment Guidelines

Environment Requirements

- **Production Server**: AWS Elastic Beanstalk
- **Memory**: Minimum 4GB RAM
- **Storage**: Minimum 32GB SSD
- **Database**: Dedicated database instance (Microsoft SQL Server)

Deployment Process

1. Deploy backend API services to **AWS Elastic Beanstalk**
2. Configure database connections (MySQL)
3. Set up environment variables within the Elastic Beanstalk environment
4. Configure domain and DNS settings via AWS Route 53 or other DNS provider
5. Perform security scanning and apply IAM role policies
6. Enable auto-scaling and load balancing if necessary