

Team: Elijah Staple, Callie Jones, Elias Ortiz, Calvin Stoehr
Team Name: Rent-A-Pup

Title: Rent-A-Pup Service

Project Summary: Puppy renting service that allows renters to rent puppies, one at a time or multiple, and after renting they can decide if they want to adopt the puppies or just return them.

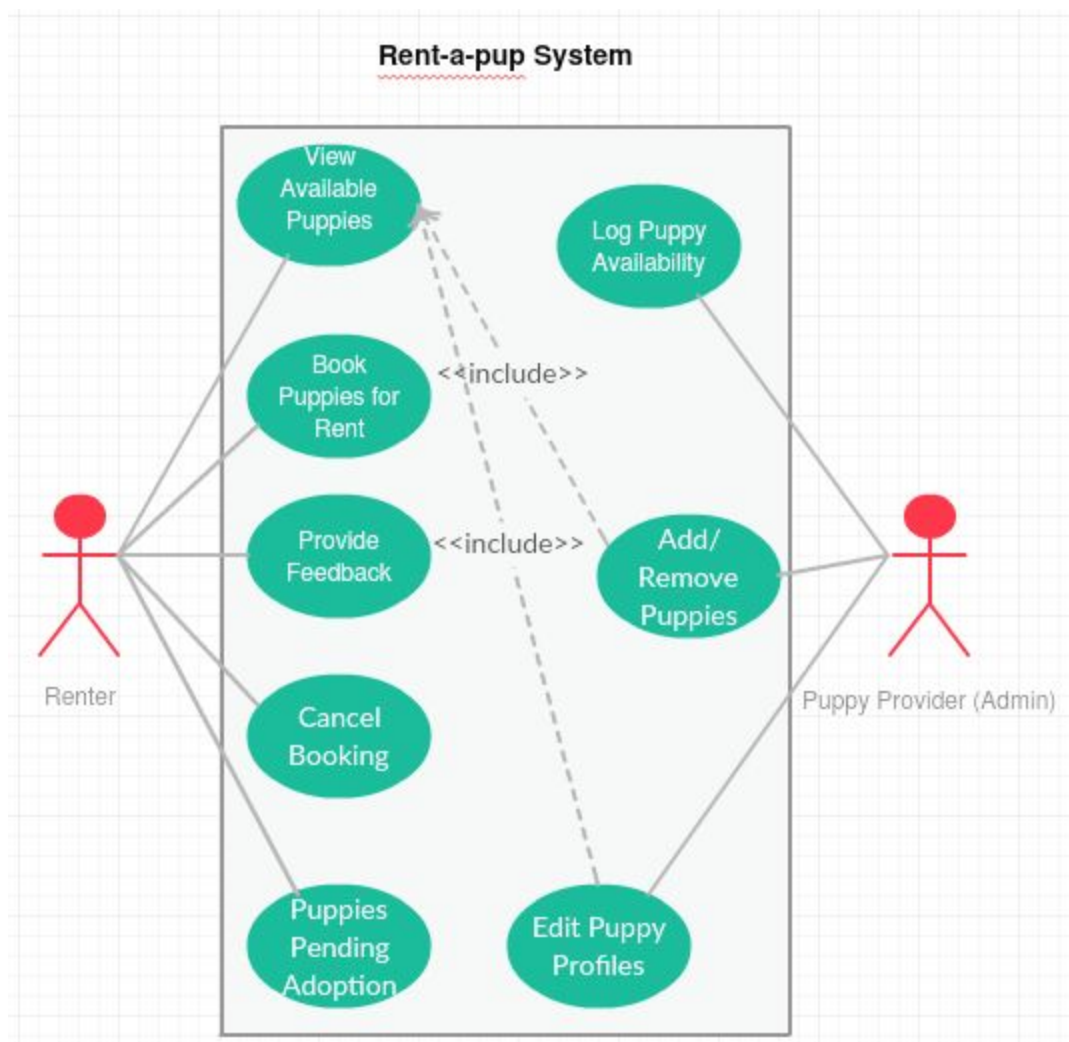
Project Requirements:

Business Requirements				
ID	Requirement	Topic Area	Actor	Priority
BR-001	Employee payments are distributed on the first of each month	Payroll	Employees	High
BR-002	Renters must provide valid payment upon placing puppy order	Finance	Renters	Medium
BR-003	Renters money is returned on a canceled booking.	Finance	Renters	Medium

User Requirements				
ID	Requirement	Topic Area	Actor	Priority
UR-001	System admin can update available puppies for rent (Add/Remove)	Updates	Admin	Medium
UR-002	Renters can view and book one or more puppies for rent	Service	User	High
UR-003	Renters can provide feedback on their experience	Service / QA	User	Medium
UR-004	Renters can cancel bookings.	Service	User	Medium
UR-005	Admin can update puppy profiles	Updates	Admin	Medium
UR-006	Users can adopt puppies at the end of the reservation period	Service	User	High

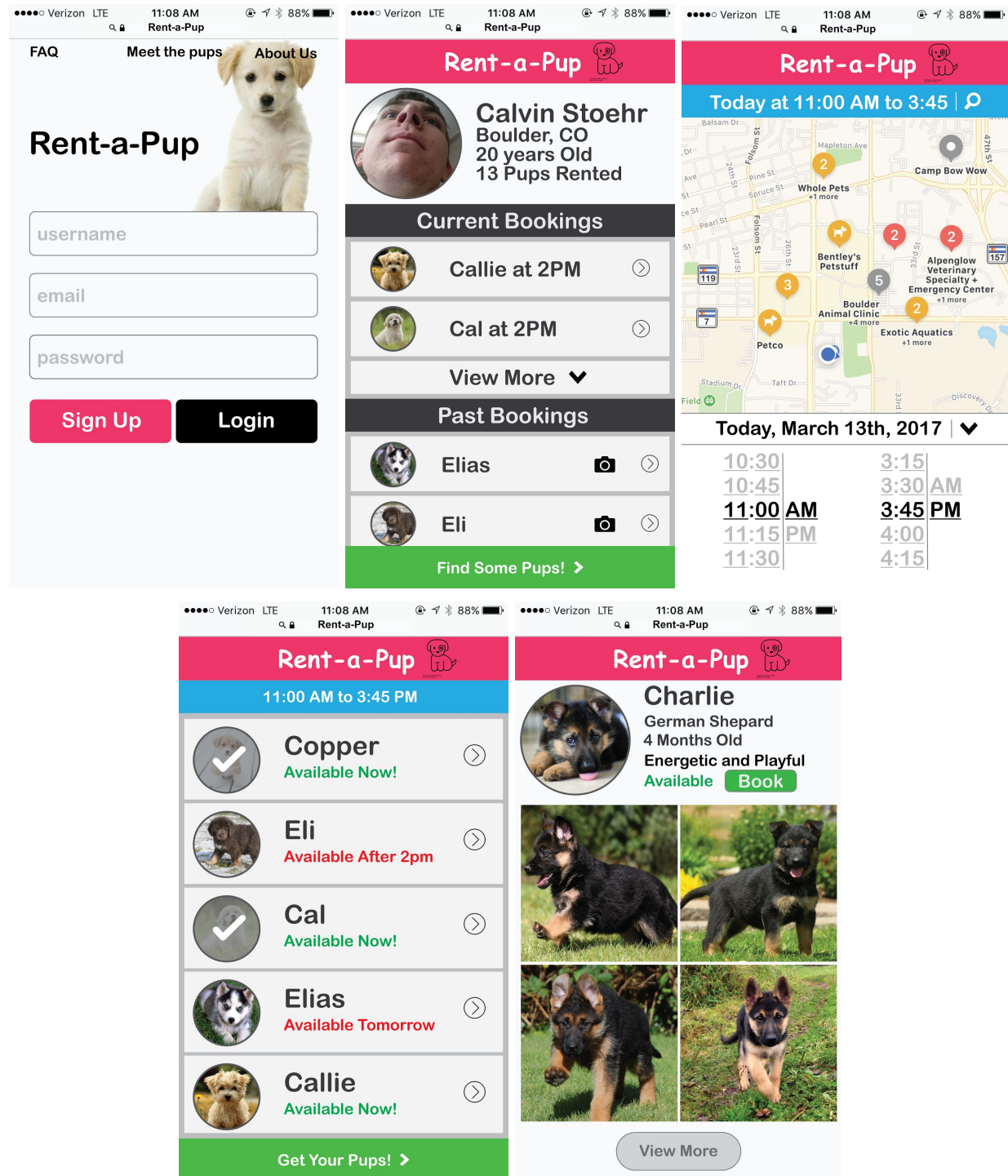
Non-Functional Requirements				
ID	Requirement	Topic Area	Actor	Priority
NFR-001	Website cannot be down on the weekends.	Reliability	System	High
NFR-002	Website must provide secure system for payment information	Security	System	Critical
NFR-003	Website must withstand heavy traffic	Durability	System	High

Use Cases:



UI Mockup:

A user will see as they work through the tasks identified in our use cases, photos of puppies, their availability and also if they are to be adopted. The overall organization of our user interface is a navigation menu that directs the user to FAQ page, the signup page and once logging in, the pups available for rent. The data is displayed in an inviting format that makes the process of renting a puppy easy. The user navigates from screen to screen, with options to explore each puppy's profile, photos and availability.



Note: Pictures are a mock of the user pages, Admin will have a few extra buttons and potentially an extra page for their specific functions.

Data Storage:

- What database are you going to use?
 - Planning on MongoDB.
- What objects do you intend to persist?
 - Puppies.
 - Accounts (Both Admin and User)
 - Eternal Life for puppies

Class Diagram:

