Project Part 2

Team: 05 Rentalit

Nicholas Clement
Joel Marquez
Brennon Lee
Christopher Struckman

A Rental Service

Title: Rentalit - A Rental Service

Project Summary: A rental service website that customers can post goods for rent, and also select goods to rent. This service will keep track of customers, the different objects for rent, and if they are available or in use currently. We will set this up for a few types of goods initially, and branch out to more, given time.

Data Storage: We are planning on storing user's login information, and the various listings associated with each user. We plan on accomplishing this through serialization of POJOs to a SQL-based database.

Project Requirements:

Business Requirements

ID	Requirement	Topic Area	Actor	Priority
BR-001	All users must have a login username consisting of first letter firstname + lastname	Authentication	All	Critical
BR-002	All users must have a password.	Authentication	All	High
BR-003	All users must be include a zip code to locate potential business.	Account Creation	All	High
BR-004	All postings must be stored in the database	Datastorage	All	Critical
BR-005	User payment information must be securely stored	Datastorage	Users	Critical

User Requirements

ID	Requirement	Topic Area	Actor	Priority
UR-001	Users can create an account	Account Creation	Users	Critical
UR-002	Admins & Users can login	Login	All	Critical
UR-003	Users must be able to search for products they want to rent	Searching	Users	High

UR-004	Users must be able to rent products from other users	Renting	Users	High
UR-005	Users must be able to post items that can be rented	Renting	Users	High
UR-006	Users must be able to rate items	Renting	Users	Medium
UR-007	Users must be able to track items currently rented out	Renting	Users	High
UR-008	Users must be able to set a rental period	Renting	Users	Medium
UR-009	Users must be able to view previous items they rented	History	Users	Low
UR-010	Users can see ratings.	Profiles	Users	Low
UR-011	Admins can search for customers by name	Admin Panel	Admins	Low

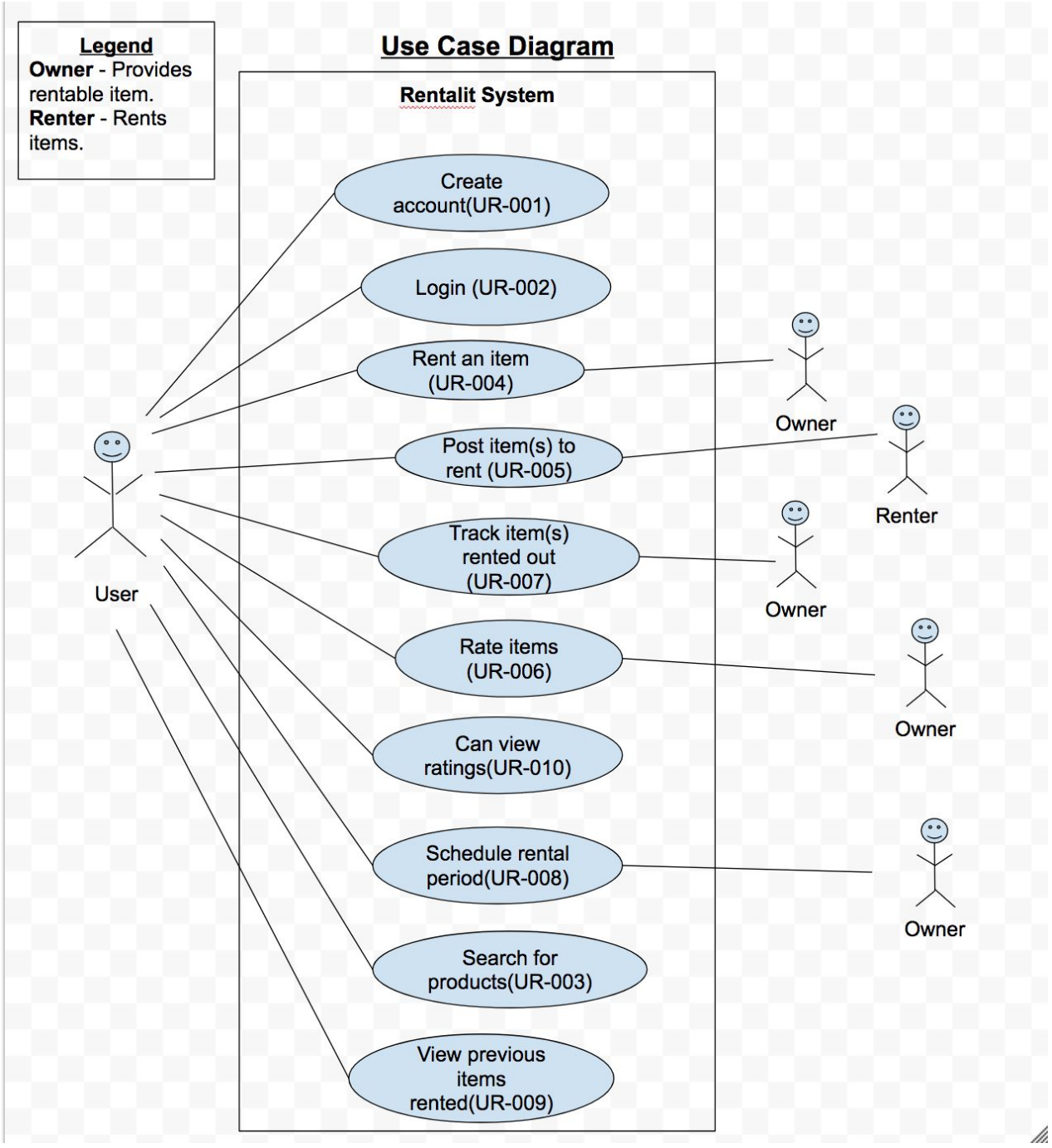
Non-Functional Requirements

ID	**Requirement**	**Topic Area**	**Actor**	**Priority**
NFR-001	All users must have a password consisting of at least one uppercase character, a number, a non-alphanumeric character, and a lower case character.	Security	All	High
NFR-002	Our service will be compatible via mobile device as well as website. (Mobile friendly)	Compatibility	Admin	Medium
NFR-003	Our system shall take no more than 300 milliseconds to respond to a user's request.	Efficiency	All	Medium
NFR-004	Our system must be extendable (	Efficiency	Admin	High

	e.g. Increasing types of objects that can be rented, adding new objects and classes)			
NFR-005	Our system must be immune to SQL injection, and Cross-Site Scripting.	Security	All	Critical

Use Cases

- ***Use Case Overview*:** Create a single overview use case diagram



here's

• ***Use Case Document(s)*:** Document how the system will support **2 major** workflow tasks via use cases documents.

User Case ID:	001
Use Case Name:	Rent an item
Description:	Search for an item to rent, book a time slot and rent item if owner approves

Actors:	Renter, Owner		
Pre-Conditions:	Renter must have Rentalit account with updated contact information. Renters must be logged in. Owner must have a valid account.		
Post-Conditions:	Renter will rent item for the specified period of time from an owner.		
Frequency of Use:	Occurs every time a Renter wants to rent an item.		
Flow of Events:		**Actor Action**	**System Response**
	1.	Renter types in search query and click search.	List of products returned (aka. search results)
	2.	Renter selects product from query results	Display rental scheduler for current product
	3.	Renter chooses rental start date, and end date (Schedule the rental period)	Notify owner
	4.	Owner accepts rental request	System stores rental period, notifies owner and rental
Variations:			
Exceptions:	Cannot find requested search item, schedule doesn't work		
Developer Notes:	Foundation of our platform		

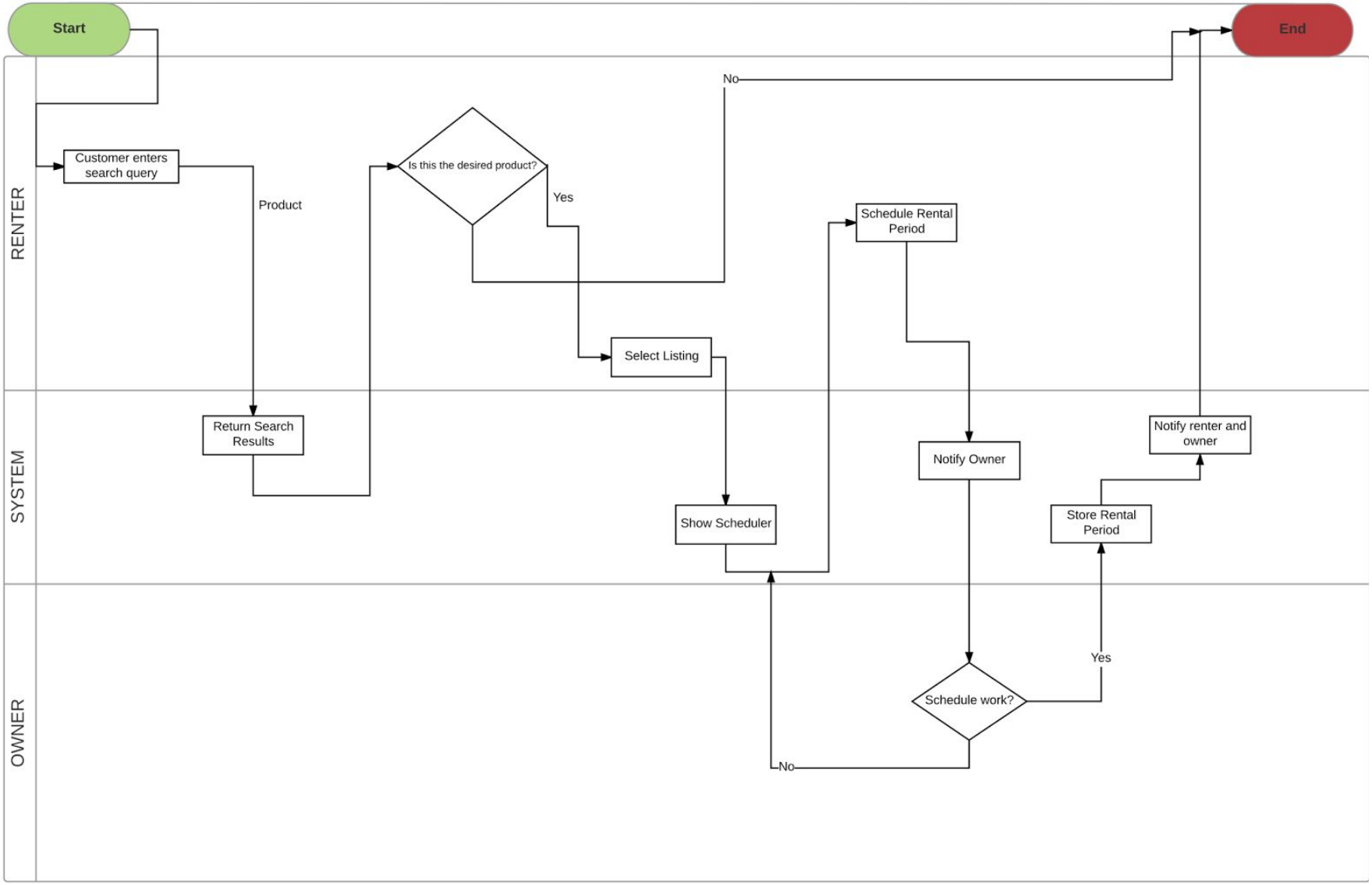
User Case ID:	UC-002
Use Case Name:	Post an item to rent
Description:	Post an item to be rented

Actors:	Owner		
Pre-Conditions:	Must have Rentalit account with updated contact information. Owner must be logged in.		
Post-Conditions:	Item is stored in product listing, and searchable by renters.		
Frequency of Use:	Every time an item is posted to be rented.		
Flow of Events:		**Actor Action**	**System Response**
	1.	Click on ‘new posting’ link	Display Posting Page
	2.	Fills out posting page and submits product	Page is submitted to the validator to check if product is valid. If valid, the product is added to listings and notifies the owner. If not, goes back to posting page to enter fields.
Variations:			
Exceptions:	Product is not valid (ie. Not rentable, missing fields, etc.)		
Developer Notes:	Essential to the renting process.		

• ***Activity Diagram:*** Activity diagram for renting:

RENT AN ITEM
USE CASE: UC-001
USER REQUIREMENT: UR-004

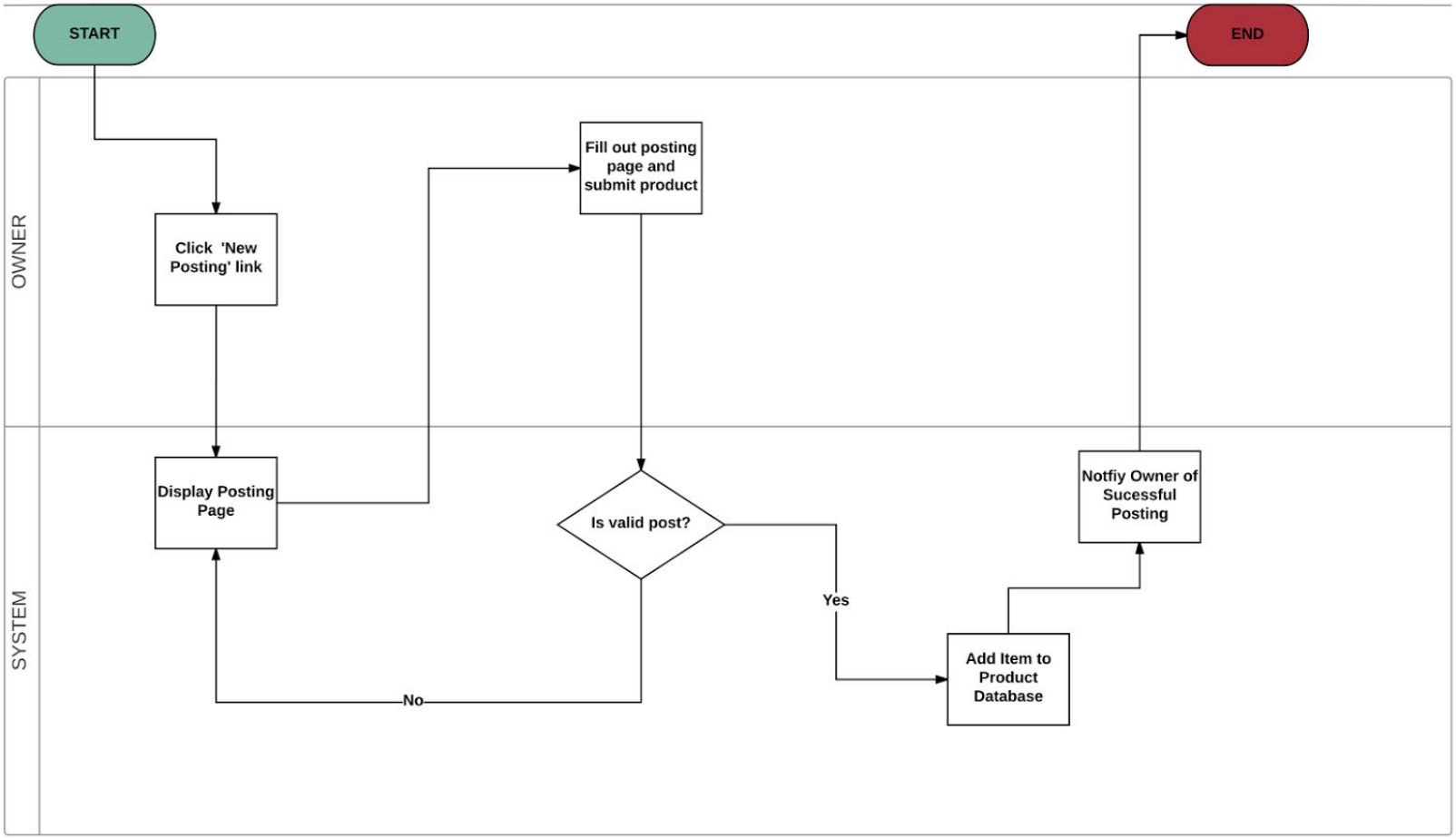
Nicholas Clement, Joel Marquez, Brennon Lee,
Christopher Struckman | October 24, 2017



POST AN ITEM FOR RENT
USE CASE: UC-002
USER REQUIREMENT: UR-005

Nicholas Clement, Joel Marquez, Nicholas Clement, Brennon Lee
October 24, 2017

Activity diagram for Posting a Rentable Item:



• ***UI Mockups:*** Create screen mockups for the user interface of various parts of your application.

https://gomockingbird.com/projects/aqkdn53

Login/Signup

Rentalit

Rentals made easy

Location

Furniture
Electronics
Tools

Rentals meeting your criteria:

	Item	Profile	Location	Rent
☐	95x95 Ad			
☐	95x95 Ad			

Make multiple pages

Description:

Lorem ipsum dolor sit amet, maiores ornare ac fermentum, imperdiet ut vivamus a, nam lectus at nunc. Quam euismod sem, semper ut potenti pellentesque quisque. In eget sapien sed, sit duis vestibulum ultricies, placerat morbi amet vel, nullam in in lorem vel. In molestie elit dui dictum, praesent nascetur pulvinar sed, in dolor pede in aliquam, risus nec error quis pharetra. Eros metus quam augue suspendisse, metus rutrum risus erat in. In ultrices quo ut lectus, etiam

Rent

User

General | Rental History | Postings

Location:

User Rating: ★★☆

Rentals:

Postings:

User

General | Rental History | Postings

- Rental 3
- Rental 2
- Rental 1

User

General | Rental History | Postings

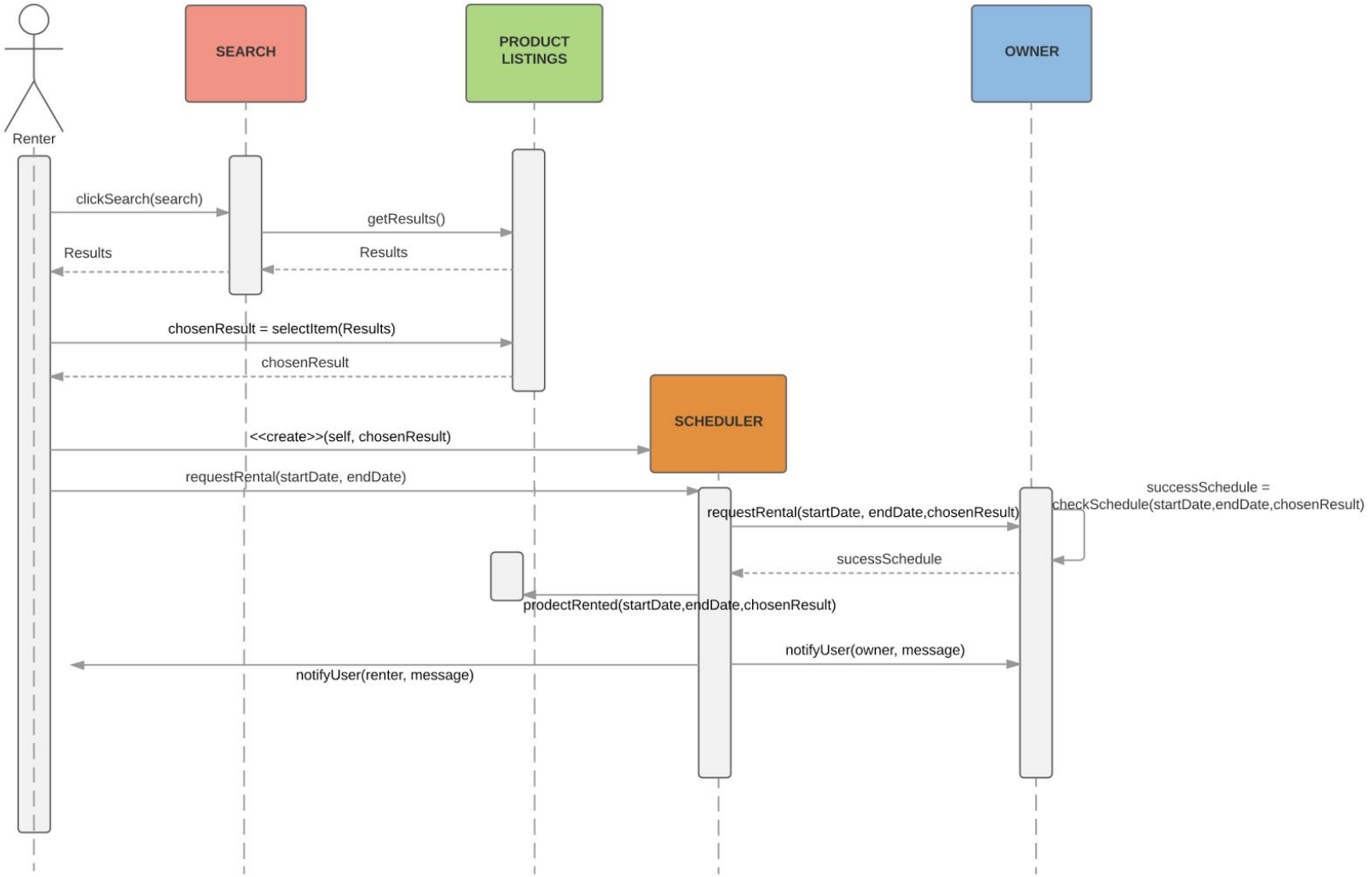
- Posting 3
- Posting 2
- Posting 1

Post New Rental

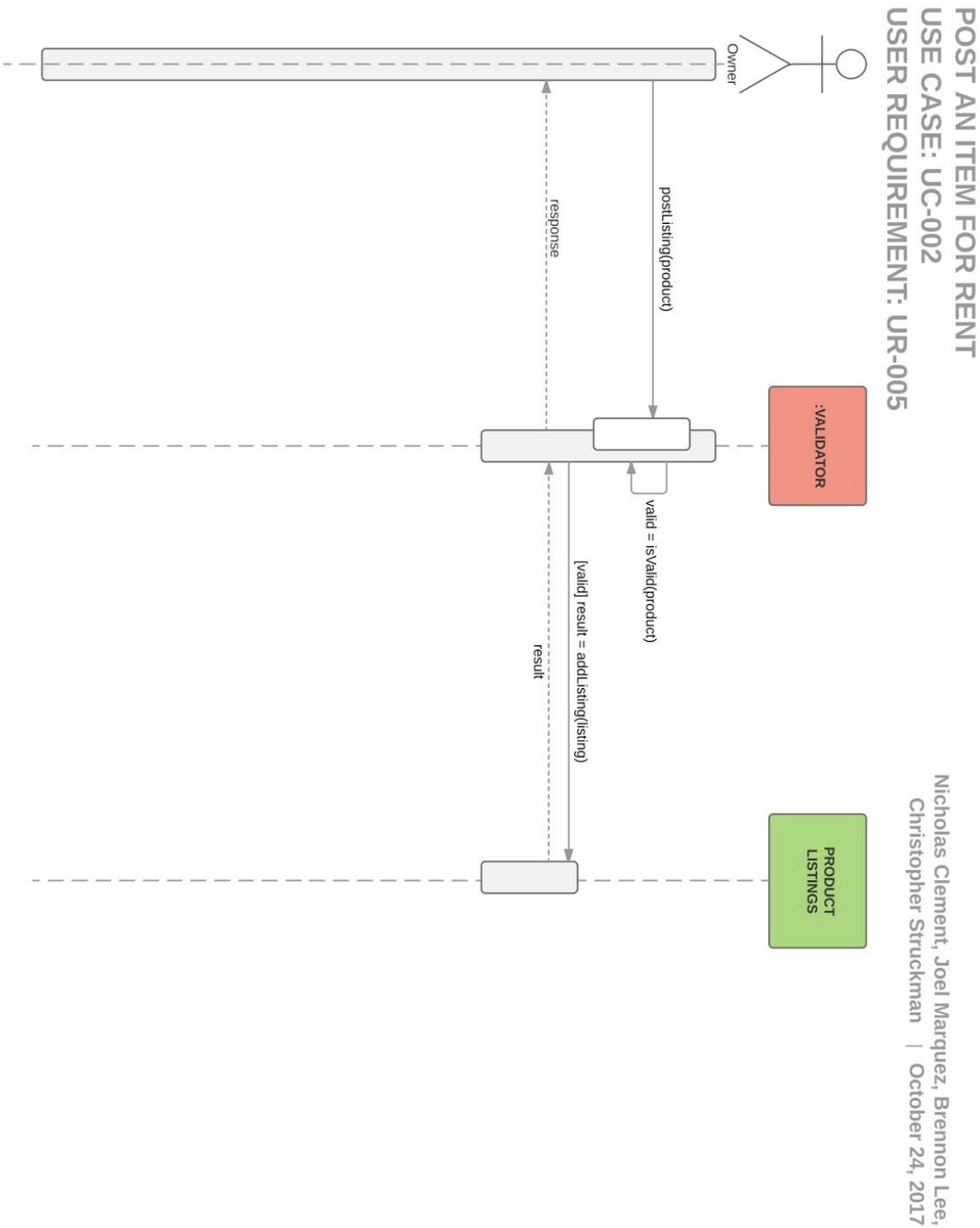
• **User Interactions:**
Rent An Item
https://www.lucidchart.com/invitations/accept/d6c4eafb-b5d9-4f54-b77b-5247f52e64b0

RENT AN ITEM
USE CASE: UC-001
USER REQUIREMENT: UR-004

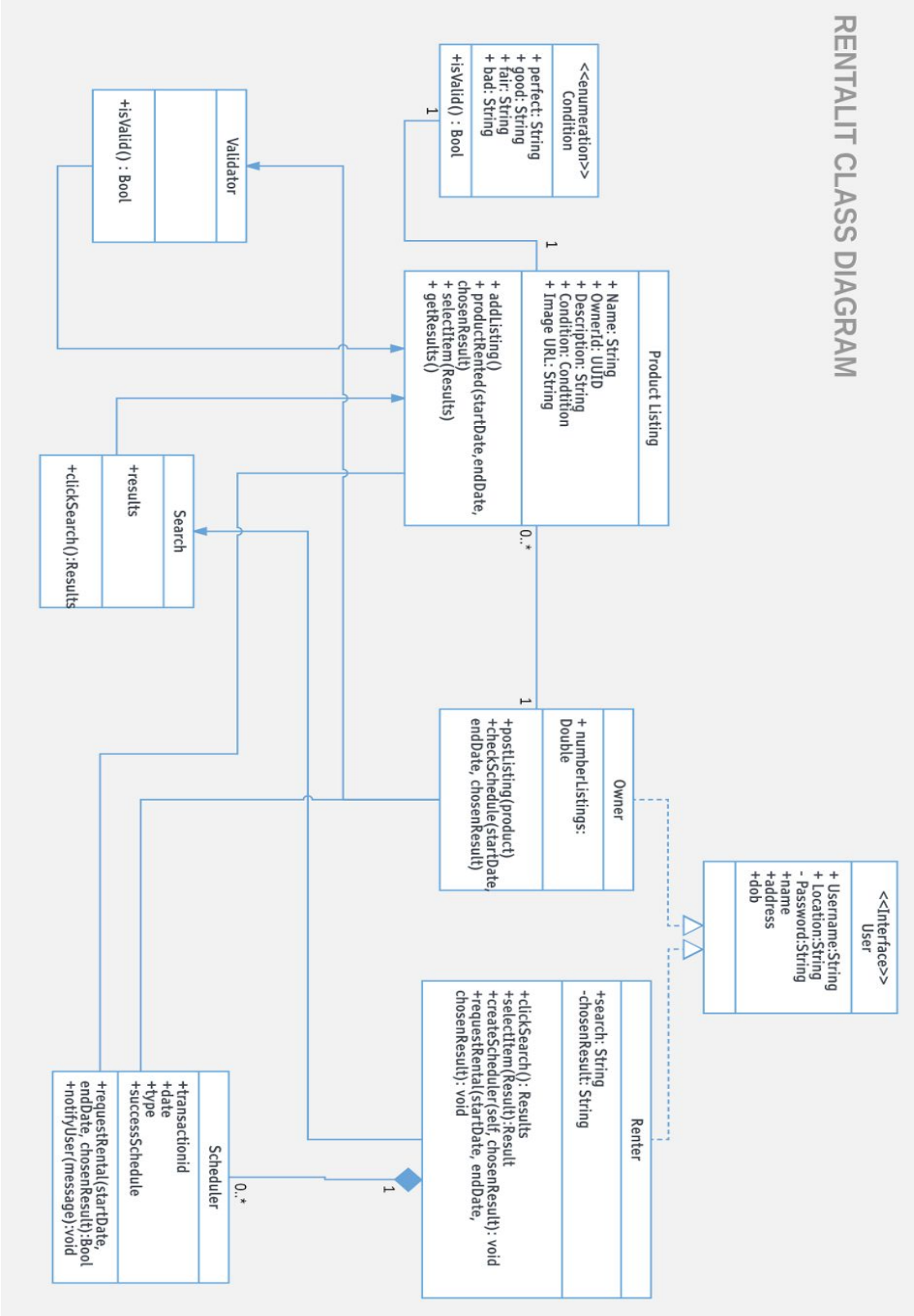
Nicholas Clement, Joel Marquez, Brennon Lee,
Christopher Struckman | October 24, 2017



Post an Item:
https://www.lucidchart.com/invitations/accept/c472bfe7-603c-4c2c-8ea9-df08af063574



• **Class Diagram:**
https://www.lucidchart.com/invitations/accept/6c4a4780-786a-4f16-8533-82848c5ebd01



Assessment

In general, you will be graded as a group on the thoughtfulness and completeness of answering the questions above, along with the overall quality of the work.

Submission

Submit a single .PDF document named ProjectGroup#_TitleOfYourProject_Part2.pdf (e.g., 03_DeliveryService_Part2.pdf) to your GitHub containing all of your work (do not put in a subdirectory).