

Guttershipe (0.3)

***Software Requirement
Specifications***

1. What is the site/app?

Guttersnipe is a web portal and mobile app that caters to anarcho-communist street youth (and adults) who desire to subvert capitalism by sharing resources.

It will enable people to broadcast to each other locations of abandoned properties that can be squatted as well as dumpsters which can be dived for food and other resources.

Eventually, other types of resource sharing will be integrated into this product. A calendar will be used to map and schedule Really Really Free Markets and Food Not Bombs meals.

Perhaps ridesharing will also be integrated

Perhaps maps of the train system and good times for hopping

Further development will require a close study of the writings of Kropotkin and Fourier.

2. What need does this meet? or problem does it solve?

This application serves the urgent need to overthrow capitalism by helping people to self-organize outside and beyond the market of commerce.

The ultimate intention is to facilitate the creation of alternate avenues of exchange, freely organized by free individuals.

3. Who is going to go/use to this site/app?

In the current incarnation, it is mostly aimed towards the freegans gutterpunks, who live off of dumpstered food, live in squatted housing, and travel by hopping trains.

As we get a better sense on the needs of the anticapitalist community and possibilities for alternative organizing, we will expand the possibilities for anti-market resource sharing.

4. Why will they go to this site/app?

To find food, clothing, shelter, etc.

5. Why will they keep coming back to your site/app?

See above.

6. How is it different from other similar sites?

There are similar sites of various types, but many of them have certain faults.

There is a site called rideshare.com; there is a site named couchsurfer.com; there is freecycle.com, which allows the sharing of goods.

These are all laudable efforts. Some of these are marred by an underlying desire for profit. But some of them are motivated out of genuine desire to promote Mutual Aid.

The very mission of Guttersnipe.net will be to promote the organization of the lumpenproletariat and to create alternative exchanges outside of capitalism. This mission will enable Guttersnipe.net to be singularly focused on this goal.

It will thus be able to bring together whatever resources necessary for the undermining of capitalism: the various services— such as squatting, dumpster diving, hitchhiking, train hopping, resource sharing, etc — will be coordinated on a singular web portal.

In addition, there are several web portals that are dedicated towards the promotion of anarcho-communist goals..

Such sites are

- Freegan.info

- Picture the Homeless
- Squat.net
- Foodsharing (Germany)

Many of our initial design specifications will be taken from the freegan group and Picture the Homeless.

In addition, we intend Guttersnipe to be cross platform, available both via the web and as a mobile app. To my knowledge, there are not yet any apps dedicated with such a task.

7. What steps will a person go through interacting with the site/app?

Most of the various interactions will be handled using forms.

The various services offered by Guttersnipe all boil essentially boil down to two types of transaction:

1. information submission;
2. information retrieval.

One person posts about an abandoned building or a good dumpster; another person searches for such information.

There may be very many different interfaces for the reporting and retrieval. Some of the data will be entered and retrieved using forms and text inputs; some will be accessed through map interfaces; some will be accessed through calendar interfaces.

In addition, there may be some need for identity management. Some users may choose to register accounts. Others may choose to always have singular, anonymous transactions.

Practical Constraints

Security

We will have to build in a security infrastructure in the project in order to guarantee anonymity of transactions.

Tor will be used to anonymize transactions.

1. Introduction

1.1. Purpose

Guttersnipe promises to be a platform for individuals and groups to freely share resources such as food, shelter, and medicine.

This Document will detail the features of Guttersnipe, and will serve as a guide to developers, and as a legal document and users manual for prospective clients.

1.2. Scope.

1.2.1. Users

1.2.1.1. User

All System users can search the Shareables view a single Shareable, and read the rating and the comments ascribed to it.

1.2.1.2. Vagrant: This class represents all visitors to the site who has not yet signed in as a member.

The Vagrant class inherits all the properties and functionalities of the baseclass User.

All Vagrants may register to become a user, may login as a user, and may retrieve a lost username or password.

1.2.1.3. Guttersnipe: This class represents a user who has registered for an account in the system.

This class inherits all the properties and functionalities of the baseclass User.

In addition, the Guttersnipe can exercise control over its own account. The Guttersnipe may edit its own profile, and edit its own availability schedule. It may renew its membership and change its password. It may logout of its account.

The Guttersnipe may also create a new Shareable, and edit or delete a Shareable that it has created.

The Guttersnipe may rate or comment on a Shareable and may delete a previous comment.

The Guttersnipe may send messages and send the schedules of other Guttersnipes. It may read messages and schedules as well. It may block any other Guttersnipe except for a Boss.

1.2.1.4. Boss: The Boss class represents the administrative users of the System.

Bosses have all the same properties and functionality as the Guttersnipe, but their functionalities are unlimited in scope.

The Boss may edit or delete any Shareable, may delete any User, and may delete any comment.

1.2.2. Business Objects

1.2.2.1. Shareable: Each shareable is classified according to its categorization and description, its schedule, and its location.

1.3. Acronyms and Abbreviations

1.3.1. V: Vagrant

1.3.2. G: Guttersnipe [User]

1.3.3. B: Boss

1.3.4. SRS: Software Requirements Specification

1.3.5. GUI: Graphical User Interface.

1.3.6. FSM: Finite State Machine.

1.3.7. 1 DB: Database.

1.3.8. 1 ERCD: Entity-Relation Class Diagram.

1.4. References

2. Appendix A: User Interface Prototypes

2.1. Summary

3. The rest of this SRS is organized as follows:

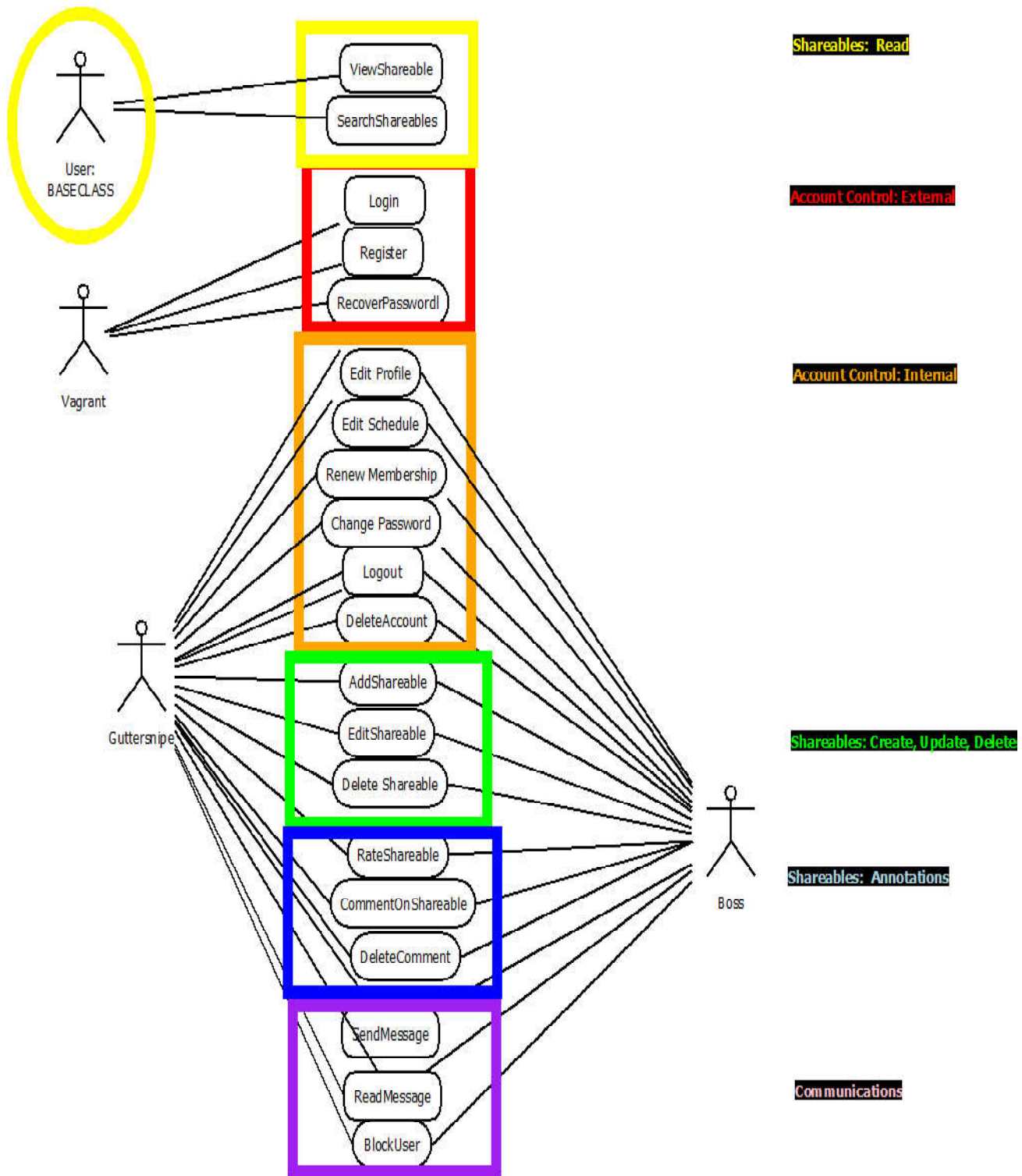
Section 2: Gives the overall description of the Guttersnipe application. It contains the Use-Case diagram and descriptions for Guttersnipe. Section 2 also contains the assumptions and dependencies of the system.

Section 3: Gives specific software requirements and functionalities in the form of Mini Use- Case diagrams along with accompanying Collaboration diagrams, Finite State Machine of the system, and ER Class diagram of the system. This section also contains supplementary software requirements of the systems.

Section 4: GUI Components: The appendix contains user interface prototypes for the system, including many screenshots from the 0.2 release of the application.

2. Overview

2.1 User Case Diagram



i. User (U)

a. Shareables (Read)

- a) ReadShareable: Any U may browse a search result.***
- b) SearchShareable: Any U may search for movies using any combination of search vectors (id, title, starts, director, year, etc.)***

ii. Vagrant (V)

a. Account Management (External)

- a) Login: Any V can login to the system, which transforms V into a G or B .***
- b) Register: Any V may register to become a G***
- c) RecoverPassword: : Any V who has forgotten account username may request that user information will be emailed***
- d) RetrieveUsername: Any V who has forgotten account username may request that user information will be emailed.***

iii. Guttersnipe (G)

a. Account Management (Internal)

- a) EditProfile: Can edit information on own account.***
- b) EditSchedule: Can edit availability schedule.***
- c) RenewMembership: Can renew the***
- d) Logout: Can log out of own account.***
- e) ChangePassword: Can reset own password from the Account Settings page.***
- f) DeleteAccount: Can delete own account.***

b. Shareables (Create, Update, Delete)

- a) *CreateShareable: Can add a Shareable resource.*
 - b) *UpdateShareable: Can update Shareable created by self.*
 - c) *DeleteShareable: Can delete Shareable created by self.*
- c. Shareables (Annotate)
 - a) *RateShareable: Can rate any shareable*
 - b) *CommentOnShareable: Can comment on any shareable.*
 - c) *DeleteComment Can delete own comment.*
- d. Communications
 - a) *SendMeetup: Can communicate message or receive meetup schedule from any other Guttersnipe.*
 - b) *ReadMessage: Can read message inbox.*
 - c) *SetSchedule: Can set daily, weekly, yearly calendar.*
 - d) *BlockUser: Can block any other user except Boss.*

iv. Boss (B)

Boss has same capabilities as Guttersnipe, but they are unrestricted to apply to all system users. The relevant overrides are as follows:

- a. **UpdateShareable: Can update Shareable created by any user.**
- b. **DeleteShareable: Can delete Shareable created by any user.**
- c. **DeleteComment Can delete Comment written by any user.**
- d. **DeleteAccount: Can delete the account of any user, except for another Boss. Deleting a user account will also delete the Boss's DeleteComments and Shareables added by the User.**
- e. **BlockUser: Triggers a DeleteAccount option for the blocked user.**

2.2 Client / Server Assumptions and Dependencies

a. Previous release

i. The previous release (0.2.1.5) of Guttersnipe was a MEAN stack application, utilizing Mongo/Mongoose, Node, Express, and Angular 1.x

ii. The deployment of a node application requires a server or PAAS that supports a node engine. We find this constraint too restrictive because many servers do not have a node engine, including the servers where we hope to do our initial deployments

iii. Although Mongo/NoSql is a fine technology, we believe that rapid, optimized database queries can best be done with an SQL database, which will obviate the need for a lot of the “middle-tier” post-processing of results from a database query

b. Current release

i. Client. The client requires a contemporary browser that supports standard HTML 5, CSS 3, and Javascript. It will use Angular 2.x as a front-end framework.

ii. The application will be ported to mobile devices initially by taking advantage of libraries like phonegap which allow one to use the device webview to deploy web interfaces. Native Android/ iOS ports may be attempted as well.

iii. For backend technologies, we sought a technology that could easily be ported to a variety of servers where no superuser access is required. We dislike bulky frameworks, preferring to add components as we need them. To this end, we have chosen the Flask / Python framework.

SPECIFIC REQUIREMENTS

Collaboration Diagrams

LOGIN

REGISTER

SEARCH SHAREABLE

VIEW SHAREABLE

ADD SHAREABLE

DELETE SHAREABLE

RATE SHAREABLE

COMMENT ON SHAREABLE

SEND MEETUP MESSAGE

READ MESSAGE

BLOCK USER

DELETE USER

DELETE COMMENT

LOGOUT

CLASS DEFINITIONS

USER CLASS DEFINITIONS

Guest User Class (extends the User Class)

Attributes:

```
*** private $userID;  
*** private $username;  
*** private $accountType; *** private $userEmail;
```

Functions:

```
*** public function login($userID, $userPassword)  
// This allows a guest user to log in to the system and become a registered user or admin  
user  
*** public function register()  
// This allows a guest user to become a registered user  
*** public function complainComment($commentID)  
// This allows a guest user to send a complaint about a comment identified by  
commentID
```

RegisteredUser Class (extends the User Class)

Attributes:

```
*** private $userID;  
*** private $username; *** private $userEmail;
```

Functions:

```
*** public function getID()  
// This returns the id of the registered user  
*** public function getUsername()  
// This returns the username of the registered user  
*** public function getEmail()  
// This returns the email address of the registered user  
*** public function complainComment($commentID)  
// This allows a registered user to send a complaint about a comment identified by  
commentID  
*** public function login($userID, $userPassword)  
// This allows a registered user to send a complaint about a comment identified by  
commentID  
*** public function rateMovie($userID, $movieID, $rating)
```

```
// This allows a registered user to give a rating to a movie.  
*** public function submitComment($userID, $movieID, $commentText)  
// This allows a registered user to post a comment to the movie page identified by  
movieID
```

BUSINESS OBJECT CLASS DEFINITIONS

Movie Class

Attributes:

```
*** private $movieID;  
*** private $movieName;  
*** private $movieYear;  
*** private $movieSummary;  
*** private $movieGenre;  
*** private $movieDirector;  
*** private $movieStars;  
*** private $movieRuntime;  
*** private $movieImageLocation; *** private $movieLocation;  
*** private $movieRating;  
*** private $moviePrice;
```

Functions:

```
*** function __construct($movieID) // Constructor for the Movie class  
*** public function playMovie($userID, $movieID)  
// Verifies that user has purchased the movie, plays movie in
```

movieLocation

```
*** public function browse()  
// Displays all information about movie  
*** public function getMovieId() // Returns movie id  
*** public function getMovieName() // Returns movie name  
*** public function getMovieYear() // Returns movie year  
*** public function getMovieSummary() // Returns movie summary  
*** public function getMovieGenre() // Returns movie genre  
*** public function getMovieDirector() // Returns movie director  
*** public function getMovieStars() // Returns movie stars  
*** public function getMovieRuntime() // Returns movie runtime  
*** public function getMovieImageLocation() // Returns file location of movie poster  
*** public function getMovieLocation() // Returns location of movie file  
*** public function watchMovie() // plays the movie  
*** public function getPrice() // Returns movie price
```

Movie Class

Attributes:

```
*** private $movieID;
```

```

*** private $movieName;
*** private $movieYear;
*** private $movieSummary;
*** private $movieGenre;
*** private $movieDirector;
*** private $movieStars;
*** private $movieRuntime;
*** private $movieImageLocation; *** private $movieLocation;
*** private $movieRating;
*** private $moviePrice;

```

Functions:

```

*** function __construct($movieID) // Constructor for the Movie class
*** public function playMovie($userID, $movieID)
// Verifies that user has purchased the movie, plays movie in movieLocation
*** public function browse()
// Displays all information about movie
*** public function getMovieId() // Returns movie id
*** public function getMovieName() // Returns movie name
*** public function getMovieYear() // Returns movie year
*** public function getMovieSummary() // Returns movie summary
*** public function getMovieGenre() // Returns movie genre
*** public function getMovieDirector() // Returns movie director
*** public function getMovieStars() // Returns movie stars
*** public function getMovieRuntime() // Returns movie runtime
*** public function getMovieImageLocation() // Returns file location of movie poster
*** public function getMovieLocation() // Returns location of movie file
*** public function watchMovie() // plays the movie
*** public function getPrice() // Returns movie price

```

Cart Class

Attributes:

```

*** private $totalPrice;
*** private $checkoutCart = array();

```

Functions:

```

*** public function getAmountItems()
// Returns the length of $checkoutCart

```



```

*** public int getCurrentTotal()
// Returns the current total price of items in the cart
*** public function removeFromCart($movie) // Removes a movie from the checkout
cart
*** public int checkout($userID, $allMovieIDs)
// Executes purchase of all movies by the user.

```

Comment Class

Attributes:

```
private $commentID; private $commentText; private $userName; private $timestamp;
```

Functions:

```

*** function __construct($commentID, $commentText, $userName, $timestamp) //
Constructor for the Comment class
*** public int getCommentID() ***
// Returns the ID of the comment
*** public function getCommentText() // Gets the text of the comment
*** public function getUserName()
// Gets the name of the commenter
*** public function getTimestamp() // Gets the date of the commenter

```

Complaint Class

Attributes:

```
*** private $complaintID; *** private $commentID; *** private $reason;
```

Functions:

```

*** function __construct($complaintID, $commentID, $reason) // Constructor for the
Complaint class
*** public function getComplaintID() // Returns the complaint ID
*** public int getCommentID()
// Returns the ID of the comment being flagged
*** public function getReason()
// Gets the reason for the complaint

```

Cart Class

Attributes:

```
*** private $totalPrice;  
*** private $checkoutCart = array();
```

Functions:

```
*** public function getAmountItems()  
// Returns the length of $checkoutCart  
*** public int getCurrentTotal()  
// Returns the current total price of items in the cart  
*** public function removeFromCart($movie) // Removes a movie from the checkout  
cart  
*** public int getCurrentTotal($userID, $allMovieIDs) // Executes purchase of all  
movies by the user.
```

Rating Class

Attributes:

```
*** private $movieID; *** private $average ; *** private $oneStars; *** private  
$twoStars; *** private $threeStars; *** private $fourStars; *** private $fiveStars;
```

Functions:

```
*** function __construct($movieID) // Constructor for the Rating class  
*** public function getAverage () // Returns the average rating  
*** public int getOnes()  
// Returns the number of one star ratings  
*** public int getTwos()  
// Returns the number of two star ratings  
*** public int getThrees()  
// Returns the number of three star ratings  
*** public int getFours()  
// Returns the number of four star ratings  
*** public int getFives()  
// Returns the number of five star ratings
```

Figure : Complaints.php

Figure :

Figure : register.php

Figure : firstLogin.php

Figure: ViewMovieInfo.php

Figure : browse.php

* Add functionality to current Classes

(
U,
GU, RU, AU, Movie, Cart
)

* Create new Object Classes

(
Player,
Comment,
Session?

All Database Returns Need Object Wrapper

)
* Create new Logic Classes

(
SearchManager, BrowseManager, MovieViewManager, LoginManager,
RegisterManager, MovieRatingsInterface, MovieProcessor, CommentManager,
CartManager, CheckoutManager, PasswordControl, LogoutControl, CommentControl,
FlaggedCommentsManager, WarningSystem, DeleteManager
)

Create new GUI Classes

(
SearchGUI,
BrowseGUI, ViewMovieGUI, LoginGUI, MainCustomerPage, MainAdminPage,
RegisterGUI, RegistrationSuccessGUI, MoviesBought, MoviePlayerInterface,
CommentGUI, ViewCartGUI,
CartGUI, CheckoutGUI, ConfirmationGUI, ResetPasswordGUI, GuestPage,
FlaggedCommentsGUI, ComplaintInterface, ComplaintsInterface, DeleteUserGUI
)

Create Database Tables

(
MOVIES,
R_USERS, RATINGS, PURCHASES, COMMENTS, COMPLAINTS

)
Create Screen Shots
(
SearchGUI,
BrowseGUI, ViewMovieGUI, LoginGUI, MainCustomerPage, MainAdminPage,
RegisterGUI, RegistrationSuccessGUI, MoviesBought, MoviePlayerInterface,
CommentGUI, ViewCartGUI,
CartGUI, CheckoutGUI, ConfirmationGUI, ResetPasswordGUI, GuestPage,
FlaggedCommentsGUI, ComplaintInterface, ComplaintsInterface, DeleteUserGUI
)

GUI COMPONENTS

(v. 0.2.1.5)

angular 1.x

bootstrap 3



create rssource

Create New Resource

Through your usage of Guttersnipe, you agree to not put yourself or any other person in legal jeopardy.

Negate

Consent

You are free to use Guttersnipe as you wish.

Create New Resource

Resource = Thing + Place + Time

Example: Free meal in Prospect Park every Wednesday from 4PM to 9PM

To report a Resource on Guttersnipe, please do the following:

1. Describe Resource
2. Map Place
3. Schedule Time
4. Confirm Resource Report

Create Resource

Create New Resource

Describe

Headline

Short explanation

You must provide a headline.

Summary

Resource Description

You must provide a summary.

Method of Access

How do I acquire resource?

Notes

Additional Notes

Confirm Description

Classify

Choose a resource type:

food

medical

housing

Classify

Type

food

Edit

Subtypes

Available Subtypes

- Free Community
- Food Donation
- Dumpster

Your Selections

Food Not Bombs

Details

Add a Detail

Island of Hawaii



Your Details

vegan

Confirm Classification

Map

Address of Resource :

Find Address

Notes

Confirm Map

Schedule

Sep 11 – 17 2016

	Sun 9/11	Mon 9/12	Tue 9/13	Wed 9/14	Thu 9/15	Fri 9/16	Sat 9/17
12pm							
1pm							
2pm							
3pm							
4pm							
5pm							
6pm							

Schedule:

None

Choose Schedule

☒ Repeating Event

Event occurs every Monday

Start Time

▲	▲	
02	:	00 PM
▼	▼	

Duration

▲	▲	
03	:	30 PM
▼	▼	

Cancel

OK

Schedule

month week day

Sep 11 – 17 2016

today

< >

	Sun 9/11	Mon 9/12	Tue 9/13	Wed 9/14	Thu 9/15	Fri 9/16	Sat 9/17
11am							
12pm							
1pm							
2pm		2:00 - 3:30 Event					
3pm							
4pm							
5pm							
6pm							

Schedule:

Date	Repeats	Start Time	End Time	Delete
	Every Monday	02:00 PM	03:30 PM	✕

Notes

GET THERE EARLY!!!

SEARCH

search

Resources

List

Map

Calendar

Headline

La Bagel Delight

Thing

Type

- food

Subtypes

- dumpster

Place

104 Front St
Brooklyn, NY
11201

Time

- Every Monday
From Monday, 12:00 AM
To Monday, 2:59 AM
- Every Tuesday
From Tuesday, 12:00 AM
To Tuesday, 2:59 AM

Full Record

Created by Apr 16, 2015 1:21:24 PM by

Headline

Trader Joe's

Thing

Type

- food

Subtypes

- dumpster

Place

130 Court St
Brooklyn, NY
11201

Time

Full Record

Created by Apr 16, 2015 1:21:24 PM by

Headline

Periandria

Thing

Type

- food

Subtypes

- dumpster

Place

175 Remsen St
Brooklyn, NY
11201

Time

- Every Monday
From Monday, 12:48 AM
To Monday, 2:59 AM

Full Record

Created by Apr 16, 2015 1:21:24 PM by

Filter Resources by

Thing

Place

Time

Page 1 of 1

Resources

List Map Calendar

Headline	Thing	Place	Time	Full Record
La Bagel Delight	Type • food Subtypes • dumpster	104 Flint St Brooklyn, NY 11201	<ul style="list-style-type: none">• Every Monday From Monday, 12:00 AM To Monday, 2:59 AM• Every Tuesday From Tuesday, 12:00 AM To Tuesday, 2:59 AM	Full Record

Created by Apr 16, 2015 1:21:24 PM by

Headline	Thing	Place	Time	Full Record
Trader Joe's	Type • food Subtypes • dumpster	130 Court St Brooklyn, NY 11201		Full Record

Created by Apr 16, 2015 1:21:24 PM by

Headline	Thing	Place	Time	Full Record
Paleteria	Type • food Subtypes • dumpster	175 Remsen St Brooklyn, NY 11201	<ul style="list-style-type: none">• Every Monday From Monday, 12:45 AM To Monday, 2:59 AM	Full Record

Created by Apr 16, 2015 1:21:24 PM by

Filter Resources by

Thing Place Time

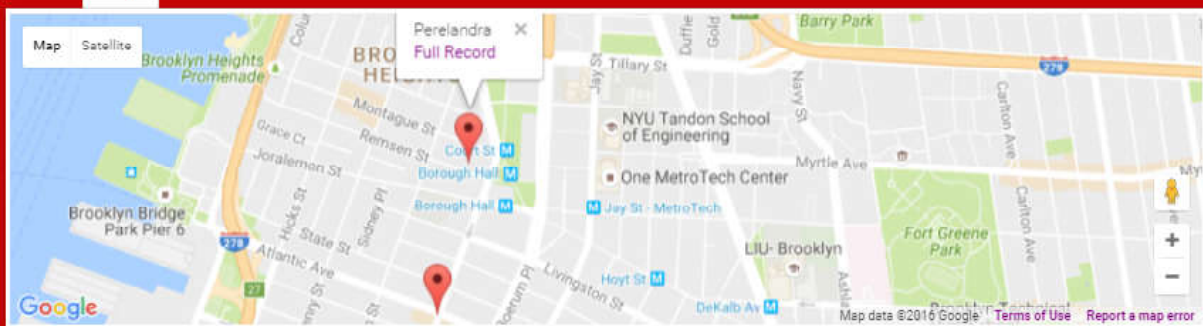
Page

Public - Resource - 1/20/16 1:21 PM

List

Map

Calendar



Show Map

[List](#)
[Map](#)
[Calendar](#)

September 2016

today
<
>

Sun	Mon	Tue	Wed	Thu	Fri	Sat
<div>1</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>2</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>3</div> <div>La Bagel Delight</div>		<div>4</div>	<div>5</div>	<div>6</div>
<div>7</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>8</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>9</div> <div>La Bagel Delight</div>	<div>10</div>	<div>11</div>	<div>12</div>	<div>13</div>
<div>14</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>15</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>16</div> <div>La Bagel Delight</div>	<div>17</div>	<div>18</div>	<div>19</div>	<div>20</div>
<div>21</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>22</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>23</div> <div>La Bagel Delight</div>	<div>24</div>	<div>25</div>	<div>26</div>	<div>27</div>
<div>28</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>29</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>30</div> <div>La Bagel Delight</div>	<div>31</div>	<div>32</div>	<div>33</div>	<div>34</div>
<div>35</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>36</div> <div>La Bagel Delight</div> <div>Perelandra</div>	<div>37</div> <div>La Bagel Delight</div>	<div>38</div>	<div>39</div>	<div>40</div>	<div>41</div>

Show Calendar

Filter Resources by

Created by Apr 16, 2015 5:25:24 PM by

Filter Resources by

Thing

Place

Time

Thing

AUTHENTICA-

[Sign Up](#)

[Sign In](#)

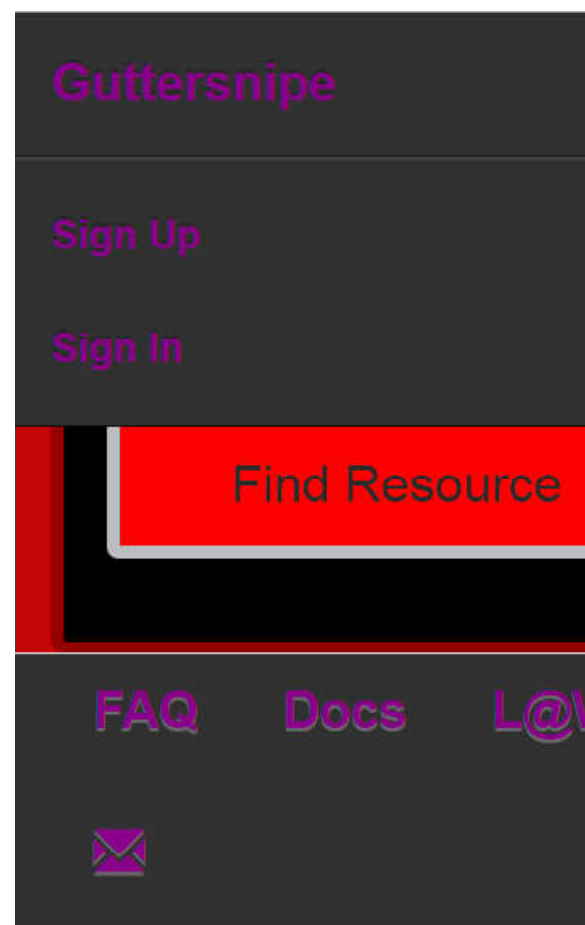
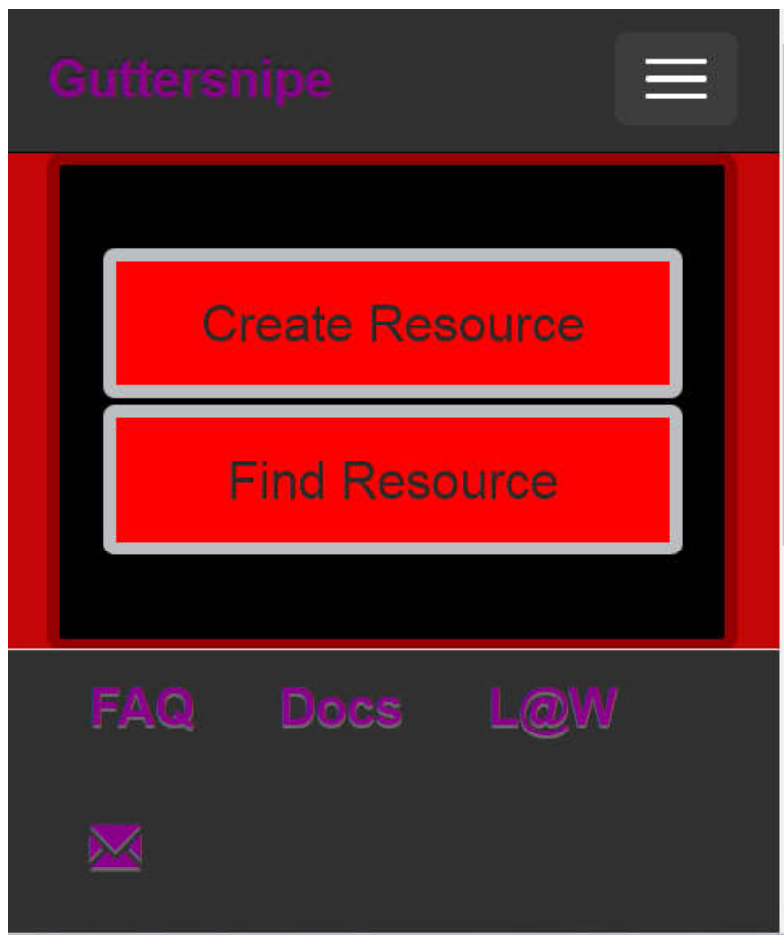
Create Resource

Find Resource

[FAQ](#) [Docs](#) [L@W](#) 

Create Resource

Find Resource



Sign up

First Name

Last Name

Email

Username

Password

[Sign up](#)

or [Sign in](#)

Username

Password

Sign in

or

Sign up

[Forgot your password?](#)

[FAQ](#)

[Docs](#)

[L@W](#)



Restore your password

Enter your account username

DOCUMENTATOIN

	FOR				
	THE				
	KIDS				

By what means could a city in a state of revolution be supplied with food? We shall answer this question, but it is obvious that the means resorted to will depend on the character of the Revolution in the provinces, and in neighbouring countries. If the entire nation, or, better still, if all Europe should accomplish the Social Revolution simultaneously, and start with thorough-going Communism, our procedure would be simplified, but if only a few communities in Europe make the attempt, other means will have to be chosen. The circumstances will dictate the measures.

The Conquest of Bread by **Peter Alexeevitch Kropotkin**

[New Chapter](#)



PROPOSAL: GUTTERSNIPE FAQ

1. What is the site/app?

- **Guttersnipe** is a web portal and mobile app that serves to search-commercial street vendors (and others) who desire to subvert capitalism by sharing resources.
- It will enable people to broadcast to each other locations of abandoned properties that can be repurposed as well as dumpsters which can be used for food and other resources.
- Eventually, other types of resource sharing will be integrated into the product. A calendar will be used to map out schedules hourly, daily, from breakfast and Food Not Bombs events.
- Perhaps interviewing will also be integrated.
- Perhaps maps of the both systems and good items for hoarding.
- Further development will require a close study of the villages of Kropotkin and Fourier.
-

2. What need does this meet? or problem does it solve?

- This application serves the urgent need to subvert capitalism by helping people to self-organize, utilize and expand the status of commons.
- The ultimate intention is to facilitate the creation of alternative sources of exchange, freely organized by those individuals.

3. Who is going to go use to this site/app?

- As the current iteration, it is mostly aimed towards the homeless guttersnipes, who live off of dumpster food, live in squatted housing, and hoard by hoarding status.
- As we get a further sense as the result of the anti-capitalist community and practitioners for alternative organizing, we will expand the possibilities for anti-capitalist resource sharing.

4. Why will they go to this site/app?

- To find food, clothing, shelter, etc.

5. Why will they keep coming back to your site/app?

- See above.

6. How is it different from other similar sites?

- There are similar sites of various types, but many of them have serious flaws.
- There is a site called ditcher.com; there is a site named couchsurfing.com; there is freecycle.com, which allows the sharing of goods.
- There are all kinds of efforts. Some of these are moved by an underlying desire for profit. But some of them are motivated out of genuine desire to promote Mutual Aid.
- The very mission of Guttersnipe.net will be to provide the organization of the kropotkinsocialist.net and to create alternative exchanges outside of capitalism. The mission will enable Guttersnipe.net to be uniquely focused on this goal.
- It will thus be able to bring together whatever resources necessary for the sustaining of capitalism: the various workers—such as squatters, dumpster diving, hoarding, food hoarding, resource sharing, etc.—will be coordinated on a single web portal.
- In addition, there are several web portals that are dedicated towards the promotion of search-commercial goals.
- Such sites are:
 - Pierogi.org
 - Mature-Free-Homeless
 - Biquini.net
 - Piercedbarring.com (Germany)

7. Notes

- Many of our initial design specifications will be taken from the freigeier.org and Pierogi.org the freigeier.org.
- In addition, we intend Guttersnipe to be cross platform, available both via the web and as a mobile app. To my knowledge, there are not yet any apps designed with such a goal.

8. What steps will a person go through interacting with the site/app?

- Most of the various interactions will be handled using forms.
- The various services offered by Guttersnipe all fall essentially fall into two types of interaction:
 - I. Information gathering.
 - II. Information release.
- One person posts about an abandoned building or a good dumpster; another person searches for such information.
- There may be very many different interfaces for the gathering and release. Some of the site will be entered and released using forms and text input; some will be accessed through map interfaces; some will be accessed through calendar interfaces.
- In addition, there may be some need for specific communication. Some such may involve the

PRESENTATION (2013)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6
Result 6 Research 6 Fin

Start

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6
Result 6 Research 6 Fin

GUTTERSNIPE

<http://www.guttersnipe.org>

Mitchell Verter



roadrunner@waste.org

March 4, 2013

Presentation for Cs59866, Senior Project

Professor Izidor Gertner



א"ת תשע"ג

NEXT

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6
Result 6 Research 6 Fin

Objective

- To overthrow capitalism by helping to establish mediums of exchange outside of the capitalist marketplace
- "Over a billion human beings live in absolute poverty, suffering from chronic malnutrition and other ills, while we have much more than an adequate material basis for a good life for all." – John Clark, *The Impossible Community Realizing Communitarian Anarchism*

[PREV](#) [NEXT](#)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6 Result 6 Research 6 Fin

Audience

- People who can not afford to or who choose not to participate in the capitalist marketplace
 - The Victimese Majority
 - Poor
 - Dislocated
 - Immigrants
 - Street people
 - Homeless
 - Traveler punks
 - Contingent/Precarious workers
 - Lumpenproletariat

PREV NEXT

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6
Result 6 Research 6 Fin

Consulting Organizations

- Freegan NYC:
 - Freegans are people who employ alternative strategies for living based on limited participation in the conventional economy and minimal consumption of resources
 - Dumpster Dive Directory
- Picture the Homeless
 - A grassroots organization founded and led by homeless people
 - Extensive list of vacant properties

[PREV](#) [NEXT](#)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6 Result 6 Research 6 Fin

Other Web Sites

- Other Websites
 - [Squat.net](#)
 - [Foodaharing \(Germany\)](#)

[PREV](#) [NEXT](#)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6 Result 6 Research 6 Fin

Question of Method

- How do you get to Carnegie Hall?

[PREV](#) [NEXT](#)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 5 A:Method 6 Example 6 Report 6 Result 6 Research 6 Fin

Answer of Method

Answer: Geolocation

- 1661 7th Ave
New York, NY 10019



[PREV](#) [NEXT](#)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6
Result 6 Research 6 Fin

Example

[PREV](#) [NEXT](#)

Presentation 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other titles 6 Q:Method 6 A:Method 6 Example 6 Report 6 Result 6 Research 6 Pin

Submit a property report

Use the following form to report information about a vacant property.

For example, to report on 1316 Park Street in San Francisco, California, 94109, type in "1316" under Number, type in "Park" under Name, choose "Street" under Suffix, type in "94109" under Zip Code.

You must fill out all fields. If you are having difficulty determining the zip code, you should try [looking at some good maps](#).

Then describe what you have found under Report.

Street Address:

Number	Name	Suffix
<input type="text"/>	<input type="text"/>	<input type="text" value="Alley"/>

Zip Code:

Zip Code
<input type="text"/>

Report:

Please describe what you have found out about the property.

<div></div>	<div></div>
-------------	-------------

Disclaimer:

One does not know who might be using this service, so the I would recommend that users avoid giving out personal information or detailing one's plans on how to use any given space.

<input type="button" value="Submit Report"/>	<input type="button" value="Reset"/>
--	--------------------------------------

[W.A.S.C. - your own squat map page](#)

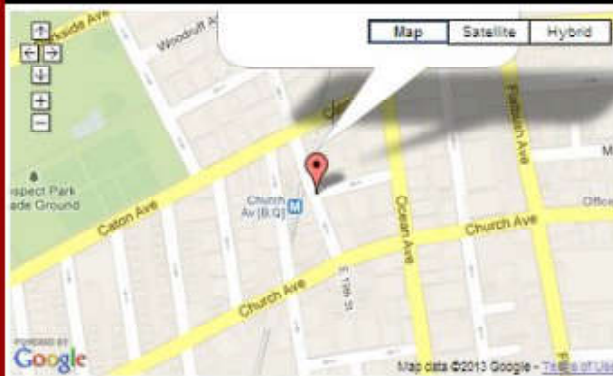
[Find vacant properties](#)

[Look at maps](#)

[Read Amanda's thoughts on this project](#)

[Privacy](#) [About](#)

Result



Property report submitted

You have submitted the following information for the address

73 st pauls pl
11226

shorty The Dog's place

PREV NEXT

Find vacant properties

Use the following form to research a vacant property and its neighborhood.

For example, to research 1315 Polk Street in San Francisco, California, 94109, type in "1315" under **Number**, type in "Polk" under **Name**, choose "Street" under **Suffix**, type in "94109" under **Zip Code**.

You must fill out all fields. If you are having difficulty determining the zip code, you should try looking at some [swell maps](#).

Then describe what you have found under **Report**.

Street Address:

Number	Name	Suffix
<input type="text"/>	<input type="text"/>	<input type="text" value="Alley"/>

Zip Code:

Zip Code
<input type="text"/>

Are you having problems determining the address? Maybe you should try looking at some [Swell Maps](#).

[W.A.S.T.E. squat, want squat main page](#)

[Report a vacant property](#)

[Look at maps](#)

[Read Kropotkin's thoughts on this project](#)

[PREV](#) [NEXT](#)

Presentation: 2013

1 Start 2 Objective 3 Audience 4 Consultations 5 Other Sites 6 Q:Method 6 A:Method 6 Example 6 Report 6 Result 6 Research 6 Fin

For Joe Strummer

"The truth is only known by guttersnipes"



<http://www.guttersnipe.org>

Mitchell Verter



[roadrunner \[at\] waste \[dot\] org](mailto:roadrunner@waste.org)


PREV

ADMINISTRATIVE

Through your usage of Guttersnipe, you agree to not put yourself or any other person in legal jeopardy.


You are free to use Guttersnipe as you wish.

All Wrongs Righted
All Rites Reversed



by

roadrunner [at] waste [dot] org



Constants

- [Guest Comment](#)
- [Profile: The Mountain](#)
- [Freeman.info](#)

John Masters

- [Web](#)
- [New Distribution](#)
- [Core Group](#)
- [New System](#)
- [Local Activities](#)
- [About Me](#)
- [Contact: Twitter](#)
- [Contact: Email](#)

#Roadrun

- [Roadrun: Twitter](#)
- [Roadrun: Email](#)

roadrunner [at] waste [dot] org



KROPOTKINS

By what means could a city in a state of revolution be supplied with food? We shall answer this question, but it is obvious that the means resorted to will depend on the character of the Revolution in the provinces, and in neighbouring countries. If the entire nation, or, better still, if all Europe should accomplish the Social Revolution simultaneously, and start with thorough-going Communism, our procedure would be simplified; but if only a few communities in Europe make the attempt, other means will have to be chosen. The circumstances will dictate the measures.

The Conquest of Bread by Pyotr Alexeyevich Kropotkin

[View Quote](#)

SCREEN SIZE ADAPTABILITY

Guttersnipe



Create Resource

Find Resource

FAQ

Docs

L@W



Guttersnipe



Sign Up

Sign In

Find Resource

FAQ

Docs

L@W



