

Product Requirements

Team f610-01f tigerPaws

*Good. Missing
Some details 9/10*

Brief problem statement

We represent a funding group (Investiny Corp.) chartered to create applications for the benefit of communities all around the country. The product we envision is called Tool Share. At its core, Tool Share is meant to enable neighbors in a community to be able to share items of common use. The successful implementation should make it easy for anyone wanting to participate to register and be able to share or borrow items.

The Tool Share product is intended to improve communities by providing an easy mechanism for sharing items between neighbors. We envision this to primarily consist of tool exchanges but the implementation must be easily extended to include items of different kinds.

We want a product whose emphasis is on ease of use, whose navigation is straightforward and where the status of items and users is clearly displayed.

We will build a web-based application that allows communities to share tools by zip code. The system will be able to be extended to various other items that can be loaned out on a temporary basis among neighbors. There will be user and tool registration and management portions of the app, community sheds that correspond to physical locations for tool storage and pickup/drop-off, and various features for accountability of tools during sharing and borrowing stages. There will also be listings and statistics to show usage of the application.

Stakeholders

Investiny Corp. Board of Directors – oversee the projects funding and expenses. Have vested interest in the proven success of the product but are not involved in the planning and execution.

Investiny Corp. Product Owner – will act as principle representative for Tool Share product needs. He/she champions the product with the Board of Directors, helps facilitate product decisions and has the ultimate say on when and what features should be released.

Software Engineering Team – is responsible for the day-to-day operations and coordination of all aspects related to the software product's life-cycle. This include, among others: planning and delegation of team roles and responsibilities; elicitation and clarification of requirements; analysis and design; implementation, testing and release of all software components.

Beta Testing Team – represent the target user base for Tools Share. Will be available in later phases of the project to conduct acceptance testing and provide feedback on product release.

Users profile

The target user must:

- have basic experience using computers and browsing the internet. Has filled out online forms or surveys and may have purchased or sold a product.
- have a computer with access to the internet

- have an interest in improving their community by lending or borrowing items of common use with others living near by
- is willing to share information such as home address and contact information

System requirements

At a high-level this project will be source controlled in SVN, run on Django using python, sqlite and needs to be compatible with the latest browsers.

Although the application needs to be accessible through the internet, deployments and demonstrations for this phase of the project will take place within the RIT Software Engineering environment. To this end, you must understand and document the target platforms from the perspective of the client browser as well as that of the server. Make sure to capture versions or software dependencies, programming languages and hardware specifications that are available for your use and proceed only after you document and confirm these with the customer.

Feature requirements (user stories)

The following list of user stories is neither final nor comprehensive. You must consider it your responsibility to maintain its relevance, clarify any misunderstandings and keep it up-to-date. Any changes must be discussed with the Product Owner for approval.

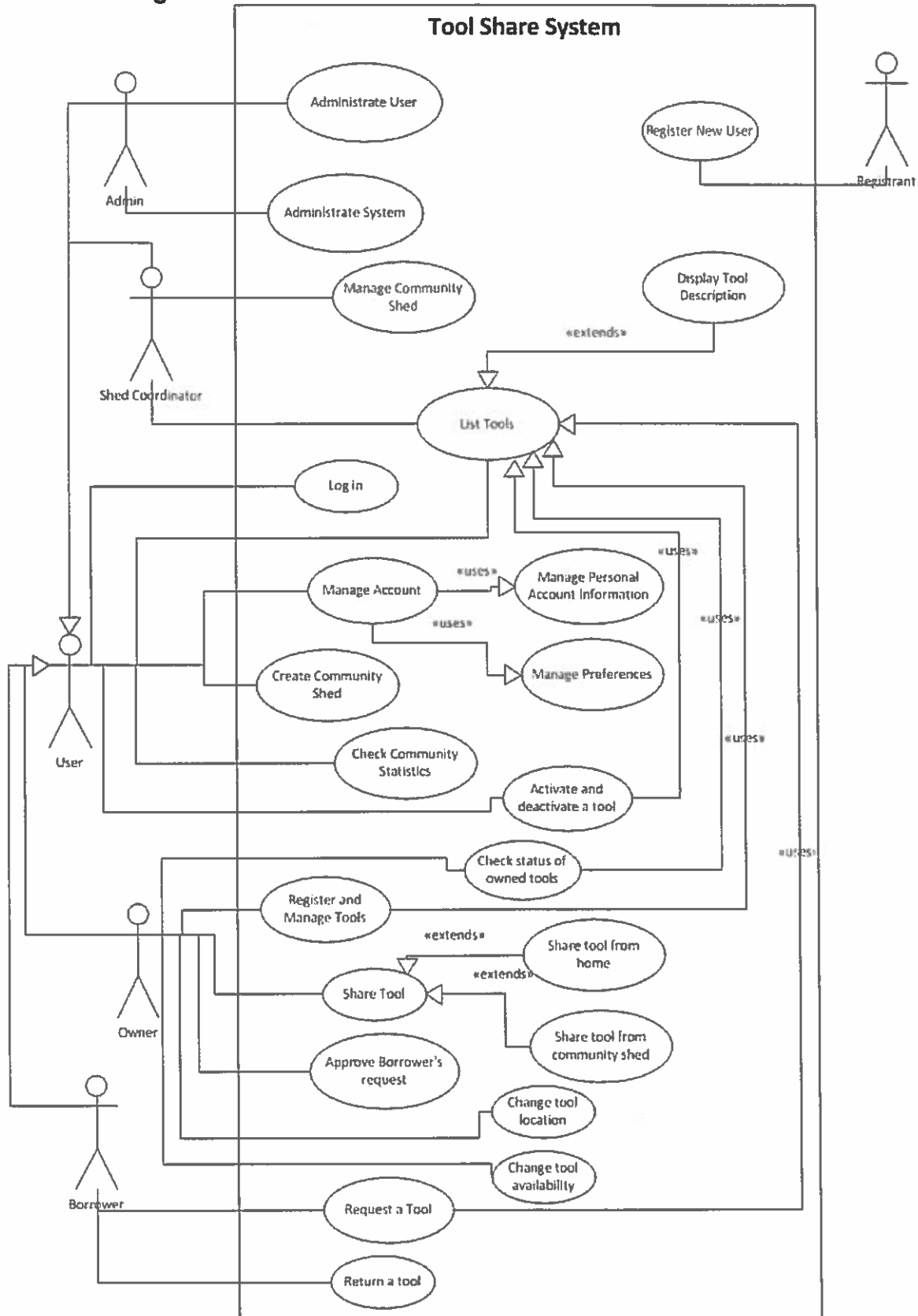
No.	User Story Name	Description	Release
1	Registration	<p>Registrant shall provide personal information and preferences to the System upon registering and becoming a User (for example: Last name or Pickup Arrangements).</p> <p>System will create a new Share Zone upon registration of User with previously unregistered zip code.</p> <p>Upon registration System will associate user with an already created Share Zone based on zip code.</p>	R1
2	Account Management - change personal info	<p>User shall be able to change personal information after registration (for example: change address).</p> <p>System will warn User if changing of community and effect any necessary changes to the item list.</p>	R1
3	Account Management - change user preferences	<p>User shall be able to change preferences after registration (for example: email reminder frequency, pickup arrangements).</p>	R1
4	Community Shed creation	<p>User shall be able to create a Shed for his Share Zone.</p> <p>A Shed has a physical location where tools from several Users can be stored.</p> <p>Shed creator becomes Coordinator of said shed.</p>	R2

5	Tool Management – registration	<p>User can register a tool by providing items information (for example: Picture, Category, Name, Description, Special instructions, etc.).</p> <p>System will require a unique field to distinguish between similar tools (for example: two laser blasters but mine has “Buzz A 113” edged on its handle).</p> <p>User must accept default Pickup Arrangement (set during User registration) or select a different arrangement for this tool (for example: “Please knock on my door” vs. “Let's make additional arrangements via email”).</p>	R1
6	Sharing – from Home	<p>User can set a previously registered tool to be shared from home.</p> <p>Tool cannot be shared from two places simultaneously.</p> <p><i>User must physically have possession of tool prior to sharing it.</i></p> <p>User can change a previously registered tool.</p>	R2
7	Sharing – from Community Shed	<p>User can set a previously registered tool to be shared from a Community Shed.</p> <p>Tool cannot be shared from two places simultaneously.</p> <p><i>User must physically move the tool to Shed prior to being allowed to share it.</i></p> <p>Shed Coordinator can, from time to time, verify and change status of tools in Community Shed (for example: a tool was returned but never marked as present or tool has gone missing).</p>	R2
8	Sharing – change location of tool	<p>User can change a previously registered tool to be shared elsewhere.</p> <p>System will prevent relocation when tool has an unresolved future reservation.</p>	R2
9	Sharing – change availability	<p>User can change availability of a tool (for example, on blackout dates when he/she will be using the tool, or be away/unwilling).</p>	R2
10	Sharing – approving a borrower	<p>System will notify User upon a request to borrow his/her tool.</p> <p>User owner of tool must approve the borrower when his/her tool is not at a Community Shed.</p> <p>System will notify borrower of decision by owner and update tools status if necessary.</p>	R2

11	Tool Listing – availability	<p>User can request from System a list of all tools and their availability. List shall be sorted by System.</p> <p>Only tools in User's Share Zone will be visible.</p>	R2
12	Borrowing – request	<p>User can select to borrow a tool for specified days provided tool is available. Borrower can add a message to request.</p> <p>User owner of tool will need to “Approve” or “Reject” the request to borrow unless tool is in a Community Shed where it is always automatically approved. Upon approval, System will create Reservation with requested dates and update availability of a borrowed tool. Upon rejection, owner of tool must provide a small reason for “Reject”.</p> <p>System will send notification email to borrower or lender including reason or message where applicable.</p>	R2
13	Borrowing - returning	<p>Borrower will return tool to pick up location unless otherwise agreed with tool owner.</p> <p><i>Borrower will notify the System that tool has been “Returned”.</i></p> <p>User owner of tool will need to notify/acknowledge to the System that tool has been “Returned” unless request to borrow was made from a Community Shed where it will be acknowledged by Shed Coordinator.</p> <p>System will update availability of a returned tool.</p>	R2
14	Tool Management – deactivation and reactivation	<p>User can deactivate a tool they previously added to the System. Neither the tool nor its reservations are shown by default, but the system still retains the information about the tool. A tool may be re-activated at any time.</p> <p>System will require confirmation in case of possible conflicts.</p> <p>System will notify borrowers of tool being deregistered and update relevant information to disable availability of tool.</p>	R2
15	Tool Management – Status	<p>User owner of tool can request from System a list of registered owned tools and their status.</p>	R2

16	Community Statistics	<p>User can request a list of statistics for their specific Share Zone. Upon request System will verify association and display such listing.</p> <p>Statistics that may be included:</p> <ul style="list-style-type: none">- most active lenders- most active borrowers- most used tools- most recently-used tools	R2
----	----------------------	--	----

Use case diagram



Use case description

Use Case Number:	UC-01
Use Case Name:	Register New User
Overview:	Registrant shall provide personal information and preferences to the System upon registering and becoming a User.
Actor(s):	User: Registrant
Pre-condition(s):	<ul style="list-style-type: none"> - System has been set up and configured. - System is running and available for registrations. - Registrant has accessed website via URL.
Scenario Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. Registrant selects option to register 2. System requests personal information 3. Registrant provided personal information. 4. System verifies required information is provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step (2) 5. System requests user's preferences 6. Registrant provides user preferences 7. System verifies required information is provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step (5) 8. If user's zip code is new to the System, new Share Zone is created 9. System will associate user with corresponding Share Zone 10. System displays confirmation of registration
	<p>Alternate Flows:</p> <p>After Step (2) System will display the option to Cancel the registration process. The following steps occur:</p> <ol style="list-style-type: none"> 1. Registrant selects option to cancel during registration 2. System requests confirmation to cancel 3. Registrant confirms intent 4. System returns to main screen
Post Condition:	Registrant did not complete registration. System does not store Registrant's information.

Registration is successful.

Use Case Number:	UC-02
Use Case Name:	Manage Personal Account Information
Overview:	<p>User shall be able to change personal information after registration (for example: change address).</p> <p>System will warn User if changing of community and effect any necessary changes to the item list.</p>
Actor(s):	User: Registered User
Pre-condition(s):	<p>- User should be registered</p> <p>- User has access to Internet</p> <p>User should have login id and password</p>
Scenario Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. User selects option to Manage Account. 2. System displays personal information. 3. User modifies personal information. 4. User Saves updated information. <ul style="list-style-type: none"> • If changes are invalid System displays message. Return to Step (2) <p>System will warn User if changing of community and effect any necessary changes to the item list</p>
	<p>Alternate Flows:</p> <p>After Step (2) System will display the option to Cancel personal information changes. The following steps occur:</p> <ol style="list-style-type: none"> 1. User clicks on Cancel button 2. System requests confirmation to cancel. 3. User confirms intent 4. User will be directed to Home Screen
Post Condition:	User successfully modifies personal information and is directed to home screen.

} specify changes clearly

Use Case Number:	UC-03
Use Case Name:	Manage Preferences
Overview:	User shall be able to change preferences after registration (for example: email reminder frequency, pickup arrangements)
Actor(s):	User: Registered User

Pre-condition(s):	<ul style="list-style-type: none"> - User should be registered - User has access to Internet - User should have login id and password
Scenario Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. User selects option to Manage Account. 2. System displays personal information and option to edit preferences. 3. User selects option to edit preferences. 4. System display user's preferences 5. User modifies preferences. 6. User Saves updated information. <ul style="list-style-type: none"> • If changes are invalid System displays message. Return to Step (4)
	<p>Alternate Flows:</p> <p>After Step (3) System will display the option to Cancel preferences changes. The following steps occur:</p> <ol style="list-style-type: none"> 1. User clicks on Cancel button 2. System requests confirmation to cancel. 3. User confirms intent 4. User will be directed to Home Screen
Post Condition:	User successfully modifies personal preferences and is directed to the home screen.

Use Case Number:	UC-04
Use Case Name:	Create Community Shed
Overview:	<p>User shall be able to create a Shed for his Share Zone.</p> <p>A Shed has a physical location where tools from several Users can be stored.</p> <p>Shed creator becomes Coordinator of said shed.</p>
Actor(s):	User :Coordinator
Pre-condition(s):	<ul style="list-style-type: none"> - User should have a shed physically. - The shed should be accessible easily by users in that particular share zone. -The shed should be big enough to accommodate tools from the users from same share zone.

Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User selects option to create shed for his zone. 2. User provides address and contact number of Shed. 3. User also gives information about availability for storage.
	Alternate Flows: User is not allowed to create shed zone ^{c-shed} if he does not meet the conditions to create shed.
Post Condition:	Coordinator ^{User} creates shed zone and is ready to share and store tools in his shed. & is made the coordinator of that shed.

Use Case Number:	UC-05
Use Case Name:	Register and manage tools
Overview:	User can register a tool by providing items information (for example: Picture, Category, Name, Description, Special instructions, etc.). System will require a unique field to distinguish between similar tools (for example: two laser blasters but mine has "Buzz A 113" edged on its handle).
Actor(s):	User: Coordinator
Pre-condition(s):	<ul style="list-style-type: none"> - Coordinator should have knowledge about tools. - Coordinator should be available when people from share zone need tools. - Coordinator should verify the quality of tools before registering it. - Coordinator should be able to manage the exchange of tools.
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. Coordinator registers tool by providing detailed information of tool in system. 2. Coordinator makes the tool available for sharing after registering it. 3. Coordinator is notified when tool is shared or returned and keeps tracks of it.
	Alternate Flows: Coordinator does not register tool because of the poor quality of tool.
Post Condition:	Coordinator initiates sharing of tool. Tool is registered & available for borrowing

Use Case Number:	UC-06
Use Case Name:	Share tool from home
Overview:	User can set a previously registered tool to be shared from home.

Actor(s):	User
Pre-condition(s):	<ul style="list-style-type: none"> - User is authenticated - User has registered a tool
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User goes to Tool Listing page 2. User selects a tool 3. System shows Tool Detail Page 4. User selects the Share Tool option 5. System requests tool location. 6. User selects Home in the Location field 7. User submits form 8. System shows Tool Shared success message
	Alternate Flows: Tool is already shared <ul style="list-style-type: none"> - If tool is already shared, show Change Location and Change Availability options instead of Share Tool option
Post Condition:	Tool is now available to be borrowed <i>from the Users home</i>

Use Case Number:	UC-07
Use Case Name:	Share tool from Community Shed
Overview:	User can set a previously registered tool to be shared from a Community Shed.
Actor(s):	User
Pre-condition(s):	<ul style="list-style-type: none"> - User is authenticated - User has registered a tool - At least one Community Shed is created
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User goes to Tools Listing page 2. User selects a tool 3. System shows Tool Detail Page 4. User selects the Share Tool option 5. User selects Community Shed in the Location field 6. User submits form 7. System shows Tool Shared success message

	Alternate Flows: Tool is already shared <ul style="list-style-type: none"> - If tool is already shared, show Change Location and Change Availability options instead of Share Tool option Community Shed doesn't exist <ul style="list-style-type: none"> - If there is no community shed created, don't include the Community Shed option in the Location field
Post Condition:	Tool is now available to be borrowed <i>from the C. shed.</i>

Use Case Number:	UC-08
Use Case Name:	Change tool location
Overview:	User can change a previously registered tool to be shared elsewhere.
Actor(s):	User
Pre-condition(s):	<ul style="list-style-type: none"> - User is authenticated - User has registered a tool - User has shared a tool
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User goes to Tools listing page 2. User selects a tool 3. System shows Tool Detail Page 4. User selects the Change Location option 5. User changes the Location field 6. User submits form 7. System shows Location Changed message <i>what if tool is currently borrowed?</i>
	Alternate Flows: Tool has not been shared <ul style="list-style-type: none"> - If the tool hasn't been shared, show Share Tool option instead of Change Location option
Post Condition:	Tool location is updated

Use Case Number:	UC-09
Use Case Name:	Change tool availability

Overview:	User can change availability of a tool.
Actor(s):	User
Pre-condition(s):	<ul style="list-style-type: none"> - User is authenticated - User has registered - User has shared a tool
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User goes to Tools listing page 2. User selects a tool 3. System shows Tool Detail Page 4. User selects the Change Availability option 5. User selects a from/to date when the tool is going to be unavailable 6. User submits form
	Alternate Flows: Tool has not been shared <ul style="list-style-type: none"> - If the tool hasn't been shared, show Share Tool option instead of Change Availability option
Post Condition:	Tool availability is updated

→ what if tool is reserved for that time period?

Use Case Number:	UC-10
Use Case Name:	Approve borrower's request
Overview:	<p>System will notify User upon a request to borrow his/her tool.</p> <p>User owner of tool must approve the borrower when his/her tool is not at a Community Shed.</p> <p>System will notify borrower of decision by owner and update tools status if necessary.</p>
Actor(s):	User: Owner
Pre-condition(s):	<ul style="list-style-type: none"> - A borrower must have already requested a tool. - A system notification has been received by the User. - User has logged in. - The tool is not in a community shed.

Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User selects option to approve borrowers 2. System show list of borrower's requests. 3. User approves request. 4. System will notify borrower of decision. 5. System will mark request as approved.
	Alternate Flows: After Step(2): <ol style="list-style-type: none"> 3. User rejects request. 4. System will ask for rejection reasons. 5. Users inputs rejection reasons. 6. System will notify borrower of decision. 7. System will mark request as rejected.
Post Condition:	System marks request as approved or rejected.

Use Case Number:	UC-11
Use Case Name:	List Tools
Overview:	User can request from System a list of all tools and their availability. List shall be sorted by System. Only tools in User's Share Zone will be visible.
Actor(s):	User: Owner, User: Borrower, User: Shed Coordinator
Pre-condition(s):	- User has logged in.
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User selects option to view tools. 2. System shows tools that belong to the user's Shared Zone. 3. System gives options for sorting and searching a certain tool. 4. User chooses sorting and search criteria.
	Alternate Flows: None
Post Condition:	User views tools availability

Use Case Number:	UC-12
Use Case Name:	Request a tool
Overview:	<p>User can select to borrow a tool for specified days provided tool is available. Borrower can add a message to request.</p> <p>User owner of tool will need to "Approve" or "Reject" the request to borrow unless tools are in a Community Shed where it is always automatically approved.</p> <p>Upon approval, System will create Reservation with requested dates and update availability of a borrowed tool.</p> <p>Upon rejection, owner of tool must provide a small reason for "Reject".</p> <p>System will send notification email to borrower or lender including reason or message where applicable.</p>
Actor(s):	User: Borrower
Pre-condition(s):	<ul style="list-style-type: none"> - User has logged in. - There must be available tools in the user's shared zone. - User has already found the tool that he wants.
Scenario Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. User clicks on a tool's "Borrow" button. 2. System requests pickup and returning date and an optional message. 3. Registrant provides the requested information. 4. System verifies the availability of the tool for the specified period of time. <ul style="list-style-type: none"> • If the tool is not available for the specified dates, System displays message. Return to Step 2 5. System displays pickup arrangement. 6. User accepts pickup arrangements. 7. System creates a reservation with requested dates and updates availability for borrowed tool. 8. System displays confirmation.

	<p>Alternate Flows:</p> <p>After Step 2 System will display the option to Cancel the request process. The following steps would occur:</p> <ol style="list-style-type: none"> 1. User selects option to cancel. 2. System requests confirmation to cancel 3. User confirms intent 4. System returns to previous screen. <p>After Step 7 if the tool is not in a community shed, the following steps would occur:</p> <ol style="list-style-type: none"> 1. System creates a borrowing request and notifies owner that he have to answer the request. 2. System displays message explaining that approval from owner is required to borrow this tool.
Post Condition:	Tool is reserved for the requested dates.

Use Case Number:	UC-13
Use Case Name:	Return a tool
Overview:	Borrower will return tool pickup location unless otherwise agreed with tool owner.
Actor(s):	Borrower, Tool owner
Pre-condition(s):	<ul style="list-style-type: none"> - System is running. - Borrower has accessed website via URL and opened the webpage of returning.
Scenario Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. Borrower logs the system and goes for returning. 2. Borrower chooses the tool which will be returned. 3. System requests borrower to return tool pickup location. 4. Borrower completes the information needed and notifies the system that tool has been "returned". <ul style="list-style-type: none"> • If request to borrow was made from a Community Shed, Shed Coordinator, instead of borrower, will notify the system. 5. System verifies that the required information is all provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step 2 6. System updates availability of a returned tool.

	Alternate Flows: After Step (2) System will display the option to cancel returning. The following steps would occur: <ol style="list-style-type: none"> 3. Borrower selects option to cancel during returning. 4. System requests confirmation to cancel 5. Borrower confirms intent. 6. System doesn't store returning information. 7. System returns to main screen.
Post Condition:	Borrower returns a tool and changes the status from unavailability to availability.

Use Case Number:	UC-14
Use Case Name:	Deactivate and reactivate a tool
Overview:	User can deactivate a tool they previously added to the System. Neither the tool nor its reservations are shown by default, but the system still retains the information about the tool. A tool may be re-activated at any time.
Actor(s):	User
Pre-condition(s):	- System is running and available for deactivation and reactivation. - User has accessed website via URL
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User logs in the system and goes to the deactivating and reactivating page. 2. User chooses to deactivate a tool. 3. System requires user to choose which tool will be deactivated. 4. System verifies required information is provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step 2 5. System requires confirmation. 6. System verifies required information is provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step 5 7. System will notify borrowers of tool being deregistered. 8. System will update relevant information to disable availability of tool.

	Alternate Flows: After Step (1) system will display the option to reactivate a tool process. The following steps would occur: <ol style="list-style-type: none"> 2. User chooses to reactivate a tool. 3. System requires user to choose which tool will be reactivated. 4. System verifies required information is provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step 2 5. System requires confirmation. 6. System verifies required information is provided. <ul style="list-style-type: none"> • If information is invalid System displays message. Return to Step 5 7. System will update relevant information to activate the tool.
Post Condition:	User deactivates or reactivates successfully. System updates relevant information.

Use Case Number:	UC-15
Use Case Name:	Check status of owned tools
Overview:	User owner of tool can request from System a list of registered owned tools and their status.
Actor(s):	User:Owner
Pre-condition(s):	- System is running and available for tool management. - User has accessed website via URL
Scenario Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. User selects option to check a list of registered owned tools. 2. System shows the list. 3. User selects one of the tools in the list to see its detailed status. 4. System verifies required information is provided. 5. System shows the selected tool's detailed status to user.
	Alternate Flows: After Step (2) System will display the option to Cancel to check detailed status. The following steps would occur: <ol style="list-style-type: none"> 3. User selects option to return back during checking. 4. User selects to go to a specific webpage of the system. 5. System goes to the specific webpage.
Post Condition:	User can check all the tools he owned and their detailed status.

Use Case Number:	UC-16
Use Case Name:	Check Community Statistics
Overview:	User can request a list of statistics for their specific Share Zone. Upon request System will verify association and display such listing.
Actor(s):	User
Pre-condition(s):	<ul style="list-style-type: none">- System is running.- User has logged in the system.
Scenario Flow:	Main (success) Flow: <ul style="list-style-type: none">1. User selects option to view statistics.2. System displays a list of statistics (including most active lenders, most active borrowers, most used tools and most recently-used tools) by verifying the association with the request.
	Alternate Flows: <ul style="list-style-type: none">1. System does not display the list of statistics.
Post Condition:	User can check community statistics.

