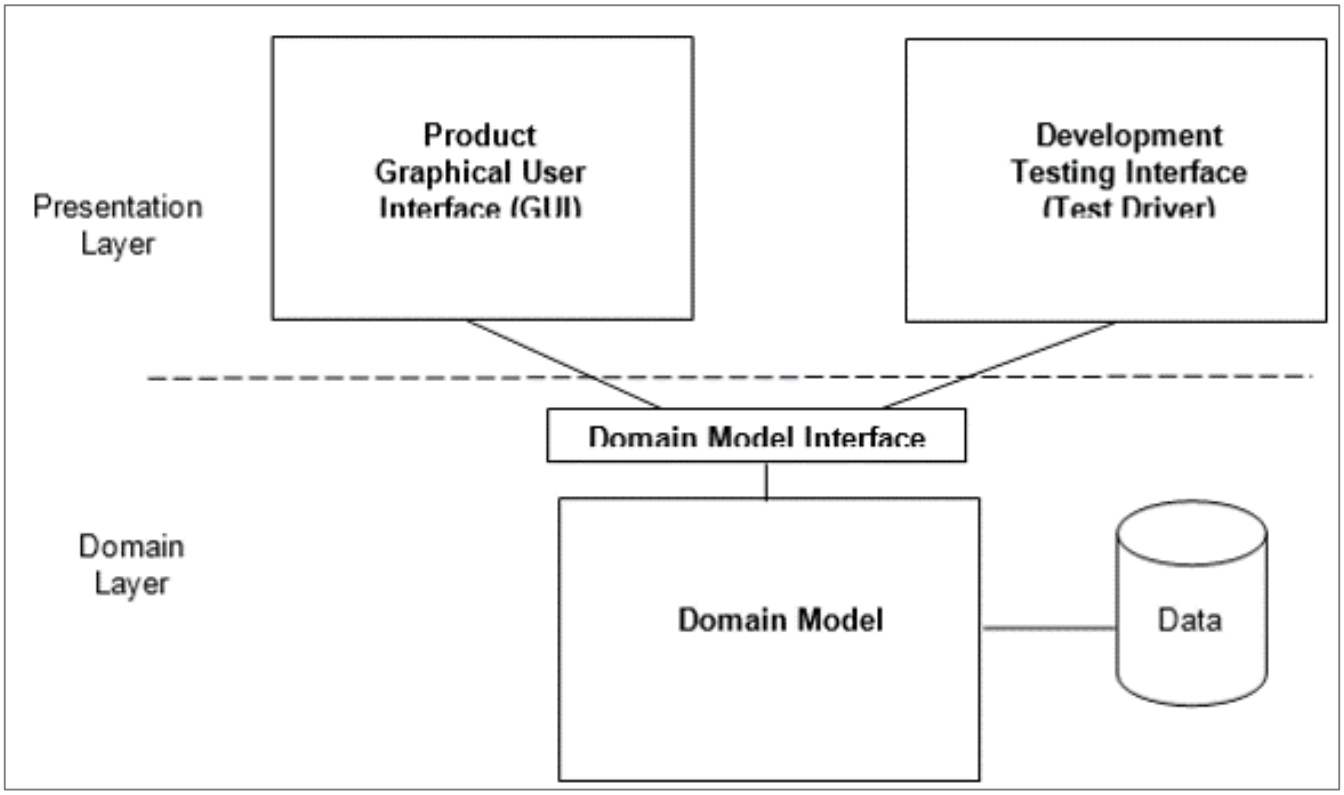
Product Design

Team X-Mates

Architectural Overview

Architectural Model



Components and Functions

User Registration	Component state ▪ List of zip codes that already exist in the system. ▪ List of valid email, password etc. Component behaviour ▪ Adds the user and its information to the list. ▪ Adds he user to registered users. ▪ Adds user to the zip code that he/she registered in. ▪ Adds the user to shed if it is needed.
Shed Creation	Component state ▪ List of sheds by the zip code. ▪ Sheds properties such as its coordinators. Component behaviour ▪ Adds the zip code to the list once the shed is created. ▪ Checks which zip code is already in use.
Tool Management	Component state ▪ Provides information about the tools used in the system for sharing. ▪ Information about the tools. ▪ Availability of the tools. ▪ Status of the tools. ▪ Owners of the tools. ▪ Borrower of a tool. ▪ List of the tools and their location. Component behaviour ▪ Provides status of the tool. ▪ Provides availability. ▪ Allows update of tool information. ▪ Allows to add new tools. ▪ Allows to change status of the tools. ▪ Allows to change availability of the tool.
Security	Component state ▪ Users' privileges. ▪ Permissions needed to access a specific page. component behaviour ▪ Checks user privileges. ▪ Allows users to access only to the pages they have permission to.

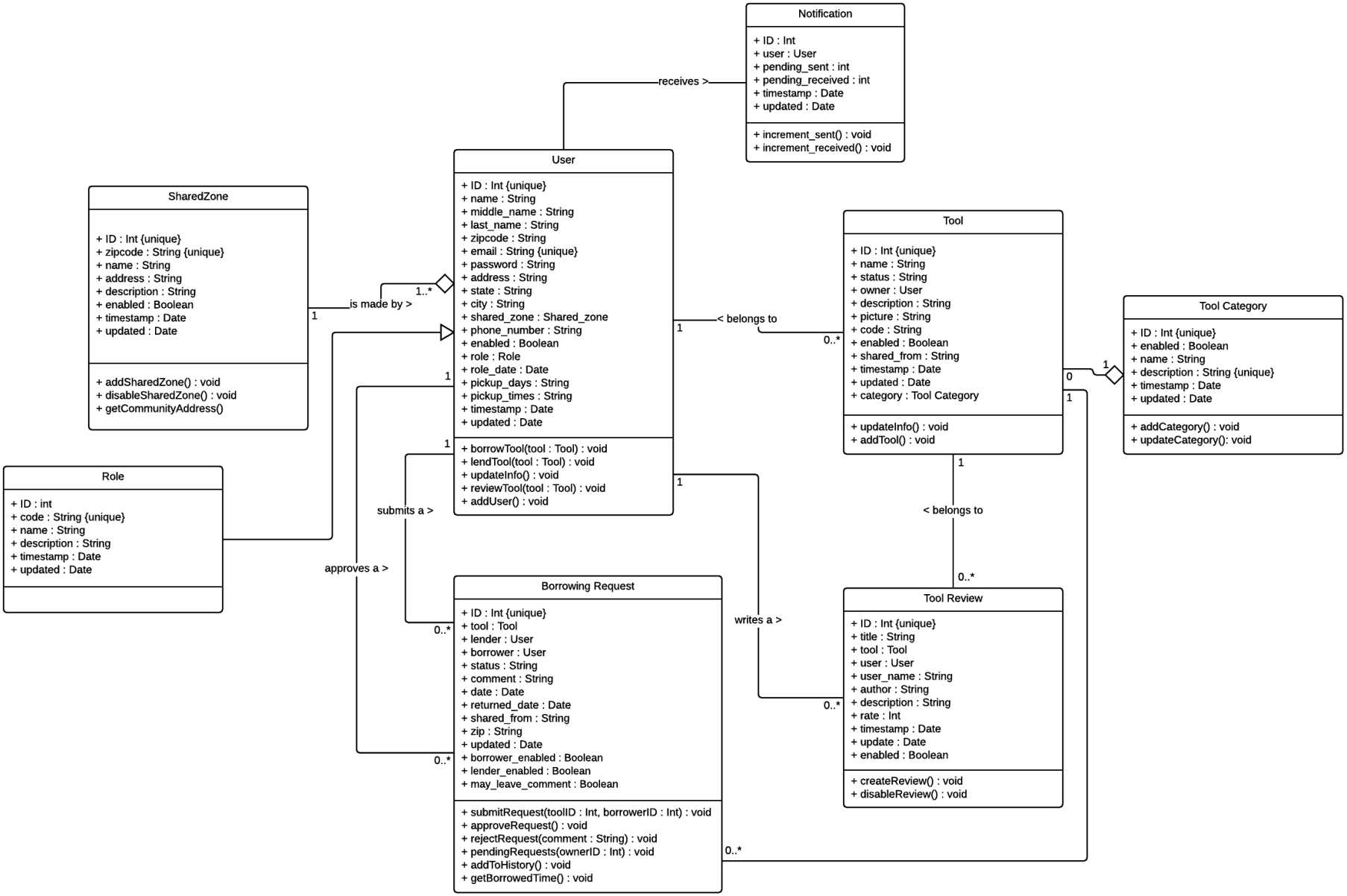
User Management	Component state ▪ Personal information about the users. ▪ Users' preferences. Component behaviour ▪ Changes user personal information such as email and password. ▪ Validates duplication of unique fields such as email. ▪ Allows creation of new users. ▪ Allows changes of users' preferences. ▪ Provides users permissions.
Review	Component state ▪ Information about reviews of a tool. ▪ Rating of tools. Component behaviour ▪ Allows creation of reviews. ▪ Allows to disable reviews. ▪ Provides rating of reviews. ▪ Provides average rating for tools.
Authentication	Component state ▪ Users' password. ▪ Users' email. Component behaviour ▪ Validates information provided to login in the system.
Requests	Component state ▪ Information about the tools asked to be borrowed. ▪ Information about the approval of tool borrowing requests. ▪ Information about rejected requests. Component behaviour ▪ Allows approval of requests. ▪ Allows creation of new requests. ▪ Allows rejection of requests. ▪ Provides information about pending requests.
Social Media	Component state ▪ Messages posted on team's 5 social media account. Component behaviour ▪ Repost messages posted on Google+ to our social media account.

Borrowing History	Component state ▪ List of previous borrowed tools. ▪ List of previous lent tools. ▪ List of borrowers who have borrowed tools. ▪ List of owners who have lent tools. Component behaviour ▪ Provides a list of previously lent tools. ▪ Provides a list of previously borrowed tools. ▪ Provides a list of users who have borrowed tools. ▪ Provides a list of users who have lent tools.
Tool Sharing	Component state ▪ List of borrowers. ▪ List of borrowers' existing tools. ▪ List of locations to share their tools. ▪ States of the shared tools. ▪ Community shed location. ▪ Community shed pickup times. ▪ Status of the available tools. ▪ Owners' sharing location. ▪ Owners' pick up times. ▪ Borrower of the tools. ▪ Owner of the tools. ▪ Shed addresses. ▪ List of available tools. ▪ Owners' information. ▪ Tools' details such as category. Component behaviour ▪ Keeps track of the status of the shared tools. ▪ Changes status of the shared tools. ▪ Updates current borrower of the shared tools. ▪ Updates information about the sharing location of the tools. ▪ Provides list of available tools. ▪ Provides details about the tools. ▪ Provides information about tools' owner. ▪ Updates availability status of the tools.

Statistic Mechanism	Component state ▪ Display statistics of the tools. ▪ Stats of previously borrowed tools. ▪ Time of borrowed tools. ▪ Most active borrowers. ▪ Most active lenders. ▪ Category of most borrowed tools. Component behaviour ▪ Bring the details of the tool status from the server and analyses it. ▪ Provides statistics to individuals about most borrowed tools. ▪ Provides statistics to admins about most active lenders. ▪ Provides statistics to admins about most active borrowers. ▪ Provides statistics to admins about most borrowed tools categories.
Tool Returning	Component state ▪ Status of the shared tools. ▪ Borrower of the tools. ▪ Owner of the tools. ▪ Returning location of the tools. ▪ Owners' addresses. ▪ Shed addresses. Component behaviour ▪ Updates availability status of the tools. ▪ Tracks borrowed tools.

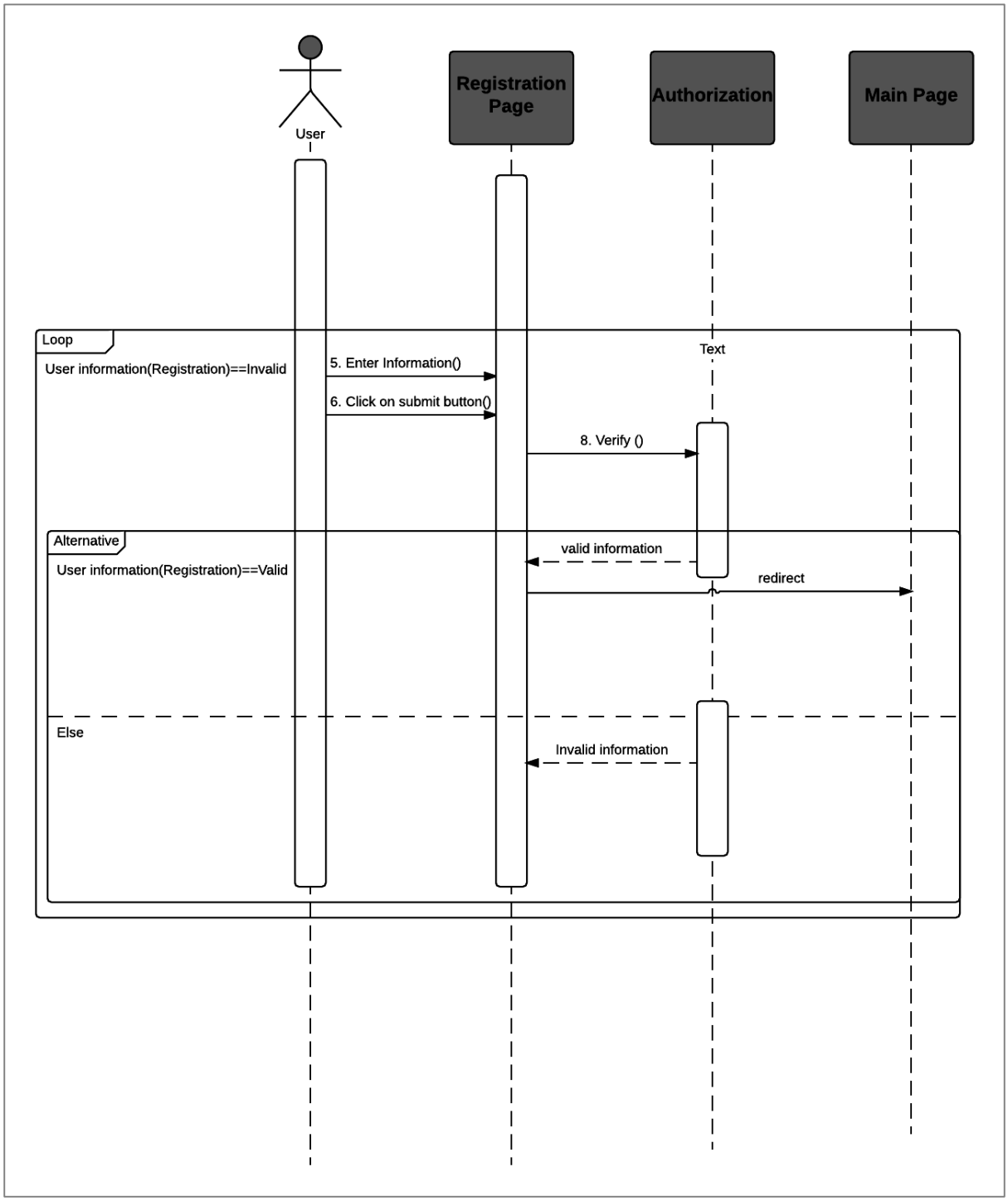
Class Diagram(s)

- **SharedZone**. Information about the shared zone. Each shared zone has a unique id, this will be the identifier to create new shared zones based on its id. It also has information (address) about the Community Shed. A Community Shed is a sharing location where users can bring their tools in case they don't want to share them at home. It will be only one per share zone and it will be created at the moment of creation of the shared zone.
- **User**. A user is a person who uses the system. They can be either borrowers or lender. They all can have tools. Users can also be admins. An admin is a super user with privileges of modifying tools, users, shared zones information, and community shed information. There are two ways a user can become an admin:
    1. The user is the first one to register with a new zip code. This action creates a new shared zone and it is automatically set as an admin.
    2. An existing admin can grant a current user with admin privileges.
- **Role**. Different user accesses to the application. It distinguishes from regular users and shed coordinators.
- **Notifications**. It contains information about the unseen notifications of the users.
- **Tool**. Information about the tools registered in the system and the owners of those tools.
- **Review**. Users can leave comments of their experience with lender and the tools they borrow. They can also leave a rating. This information helps future users to make better decisions when they want to borrow a tool.
- **Tool Category**. Different categories that a tool can be part of.
- **Borrowing Request**. List of applications submitted by borrowers. Every time a user wants to borrow a tool they have to submit a request to the owner and the owner has to approve it or rejected. This only happens if the tool's pick up location is the owners' house. If the tool's pick up location is the community shed, then the requests are approved automatically. Information about all the tool that have been borrowed. It has information about the tool, owner, borrower, date of request, date of returning.

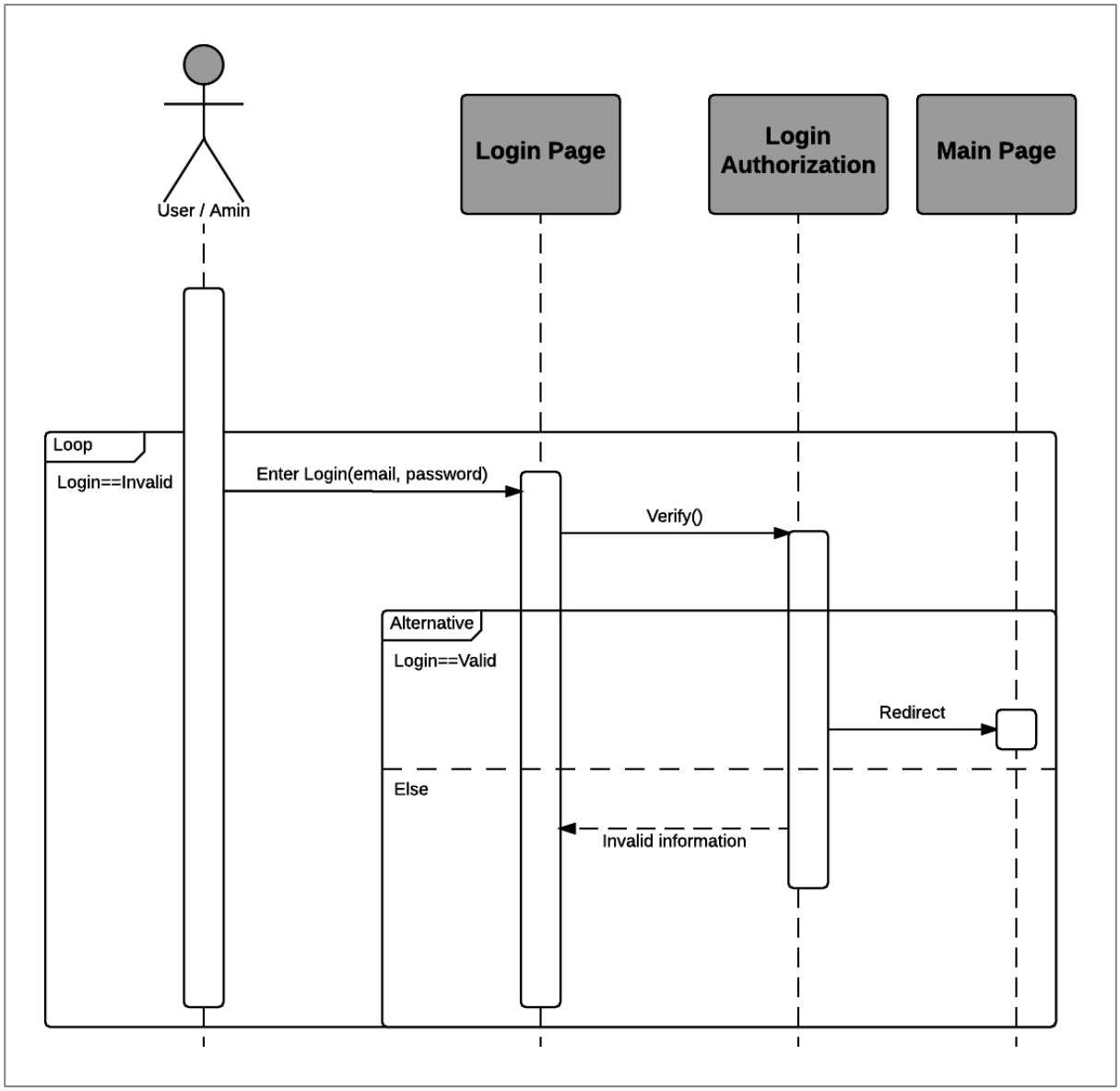

Notification
+ ID : Int
+ user : User
+ pending_sent : int
+ pending_received : int
+ timestamp : Date
+ updated : Date
+ increment_sent() : void
+ increment_received() : void
receives >
User
+ ID : Int {unique}
+ name : String
+ middle_name : String
+ last_name : String
+ zipcode : String
+ email : String {unique}
+ password : String
+ address : String
+ state : String
+ city : String
+ shared_zone : Shared_zone
+ phone_number : String
+ enabled : Boolean
+ role : Role
+ role_date : Date
+ pickup_days : String
+ pickup_times : String
+ timestamp : Date
+ updated : Date
+ borrowTool(tool : Tool) : void
+ lendTool(tool : Tool) : void
+ updateInfo() : void
+ reviewTool(tool : Tool) : void
+ addUser() : void
SharedZone
+ ID : Int {unique}
+ zipcode : String {unique}
+ name : String
+ address : String
+ description : String
+ enabled : Boolean
+ timestamp : Date
+ updated : Date
+ addSharedZone() : void
+ disableSharedZone() : void
+ getCommunityAddress()
is made by >
1..*
1
Tool
+ ID : Int {unique}
+ name : String
+ status : String
+ owner : User
+ description : String
+ picture : String
+ code : String
+ enabled : Boolean
+ shared_from : String
+ timestamp : Date
+ updated : Date
+ category : Tool Category
+ updateInfo() : void
+ addTool() : void
< belongs to
1
0..*
Tool Category
+ ID : Int {unique}
+ enabled : Boolean
+ name : String
+ description : String {unique}
+ timestamp : Date
+ updated : Date
+ addCategory() : void
+ updateCategory(): void
0
1
1
1
1
1
Role
+ ID : int
+ code : String {unique}
+ name : String
+ description : String
+ timestamp : Date
+ updated : Date
submits a >
approves a >
< belongs to
0..*
Borrowing Request
+ ID : Int {unique}
+ tool : Tool
+ lender : User
+ borrower : User
+ status : String
+ comment : String
+ date : Date
+ returned_date : Date
+ shared_from : String
+ zip : String
+ updated : Date
+ borrower_enabled : Boolean
+ lender_enabled : Boolean
+ may_leave_comment : Boolean
+ submitRequest(toolID : Int, borrowerID : Int) : void
+ approveRequest() : void
+ rejectRequest(comment : String) : void
+ pendingRequests(ownerID : Int) : void
+ addToHistory() : void
+ getBorrowedTime() : void
0..*
0..*
writes a >
Tool Review
+ ID : Int {unique}
+ title : String
+ tool : Tool
+ user : User
+ user_name : String
+ author : String
+ description : String
+ rate : Int
+ timestamp : Date
+ update : Date
+ enabled : Boolean
+ createReview() : void
+ disableReview() : void
0..*
0..*

Sequence Diagram(s)

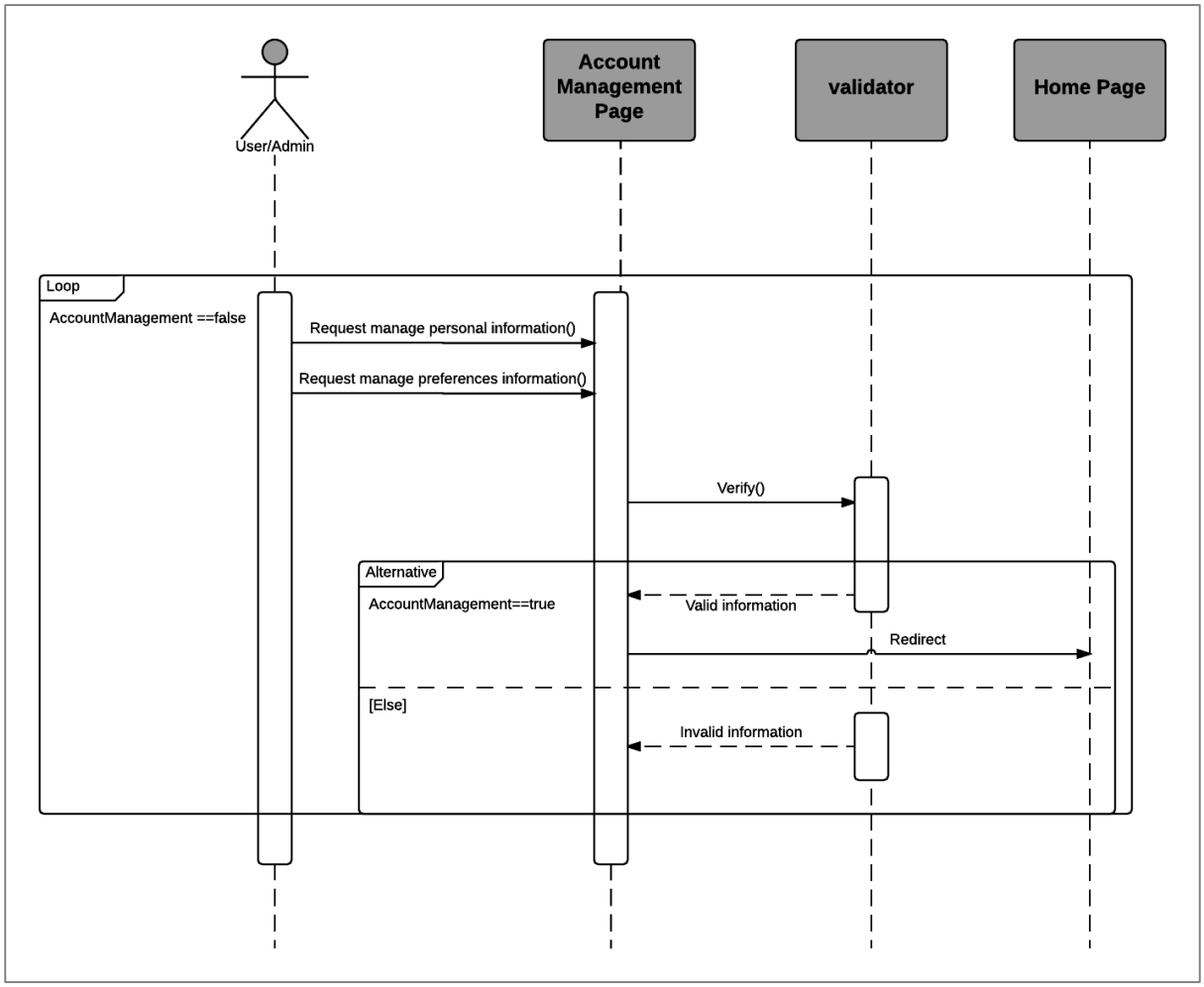
- **Registration**



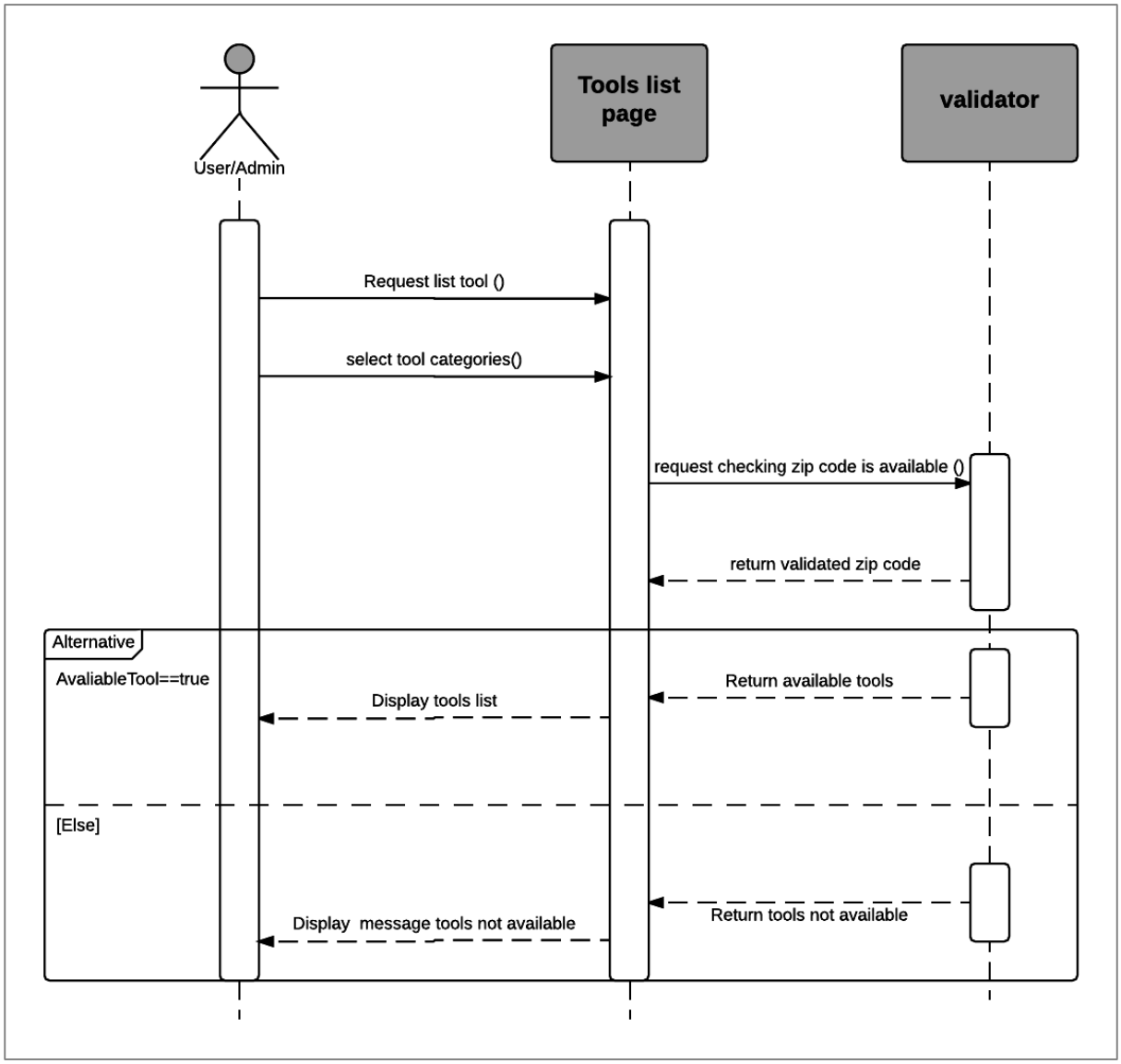
❖ **Login**


User / Amin
Login Page
Login Authorization
Main Page
Loop
Login==Invalid
Enter Login(email, password)
Verify()
Alternative
Login==Valid
Redirect
Else
Invalid information

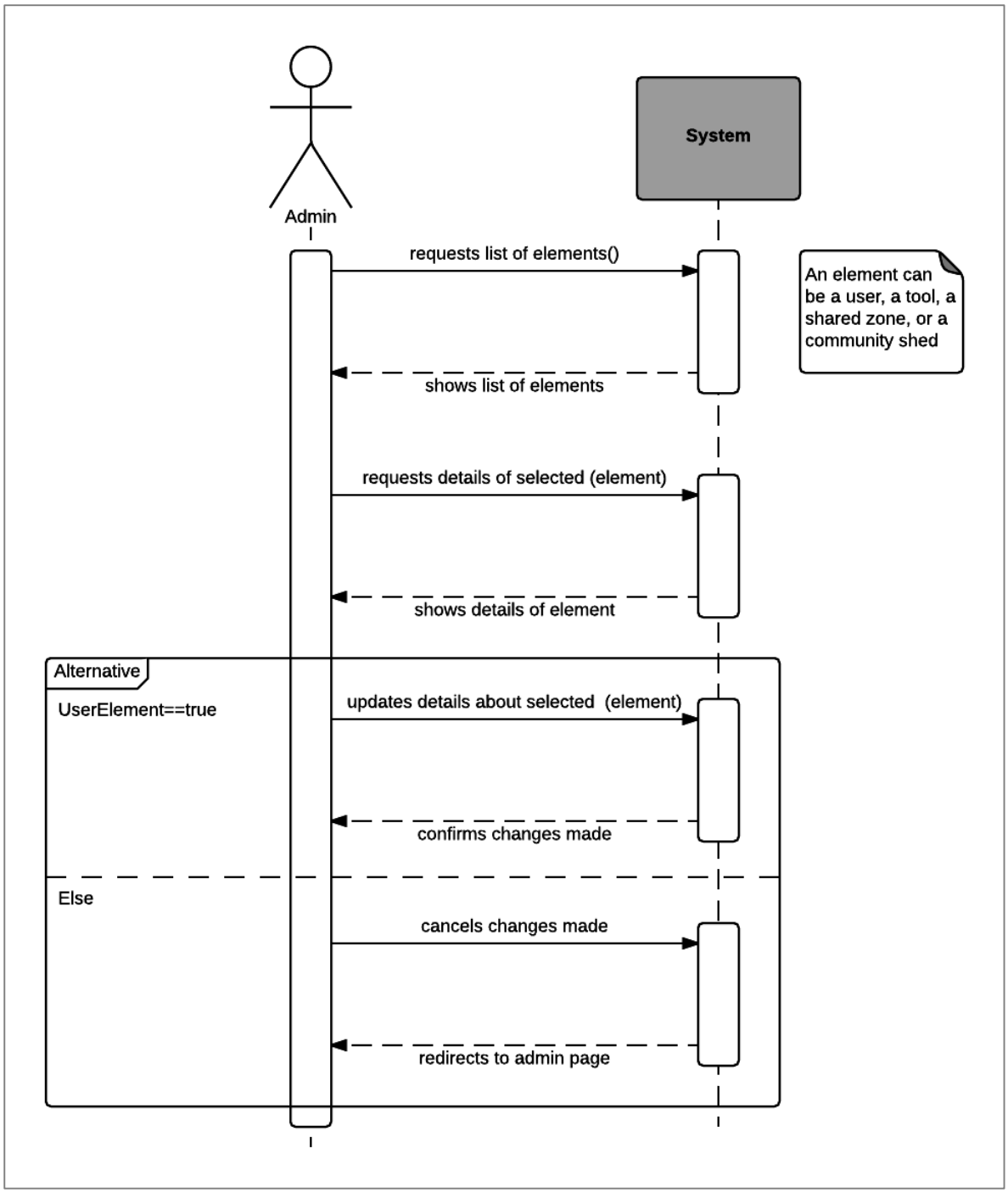
❖ Account Management



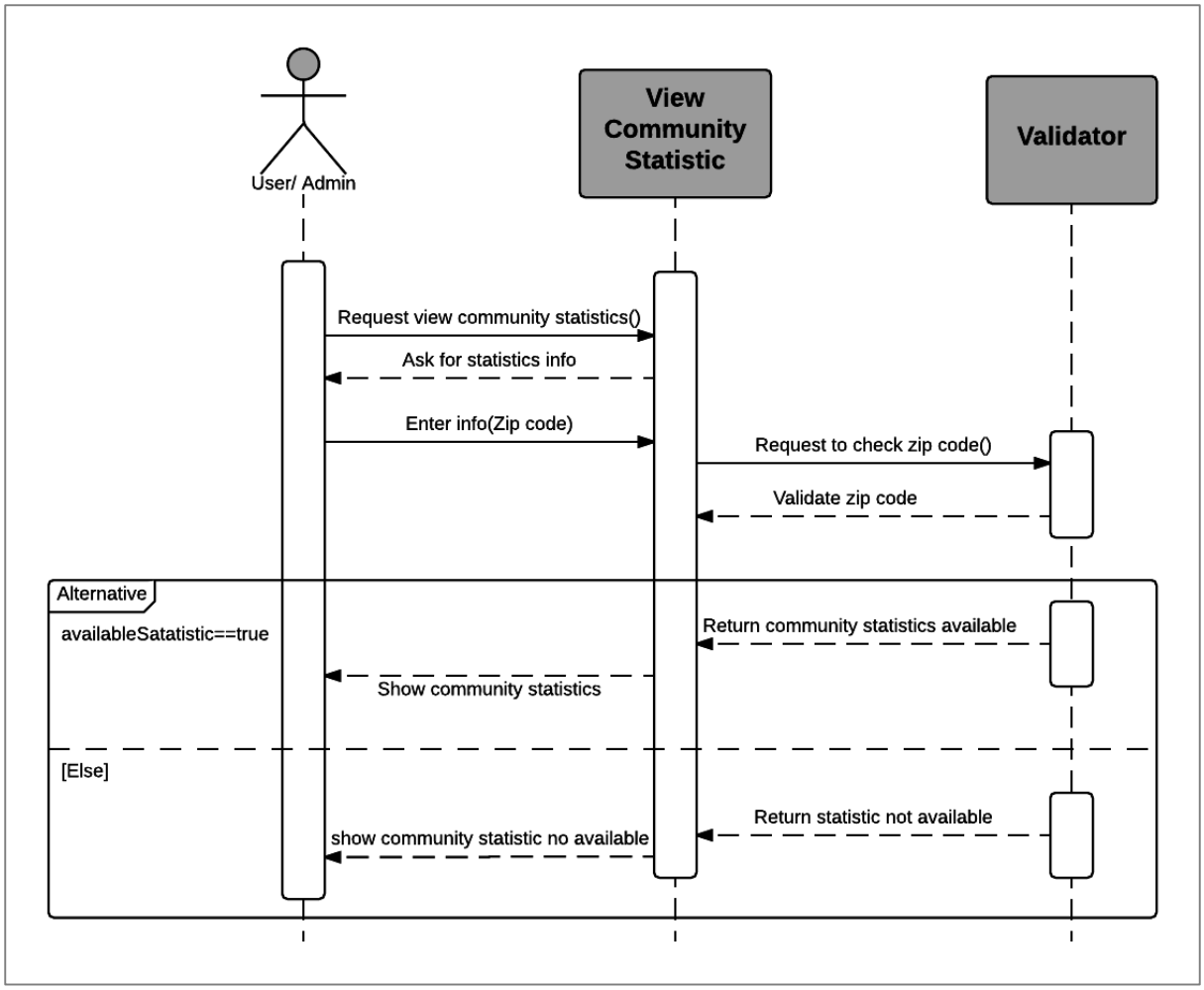
❖ **Tool Listing**



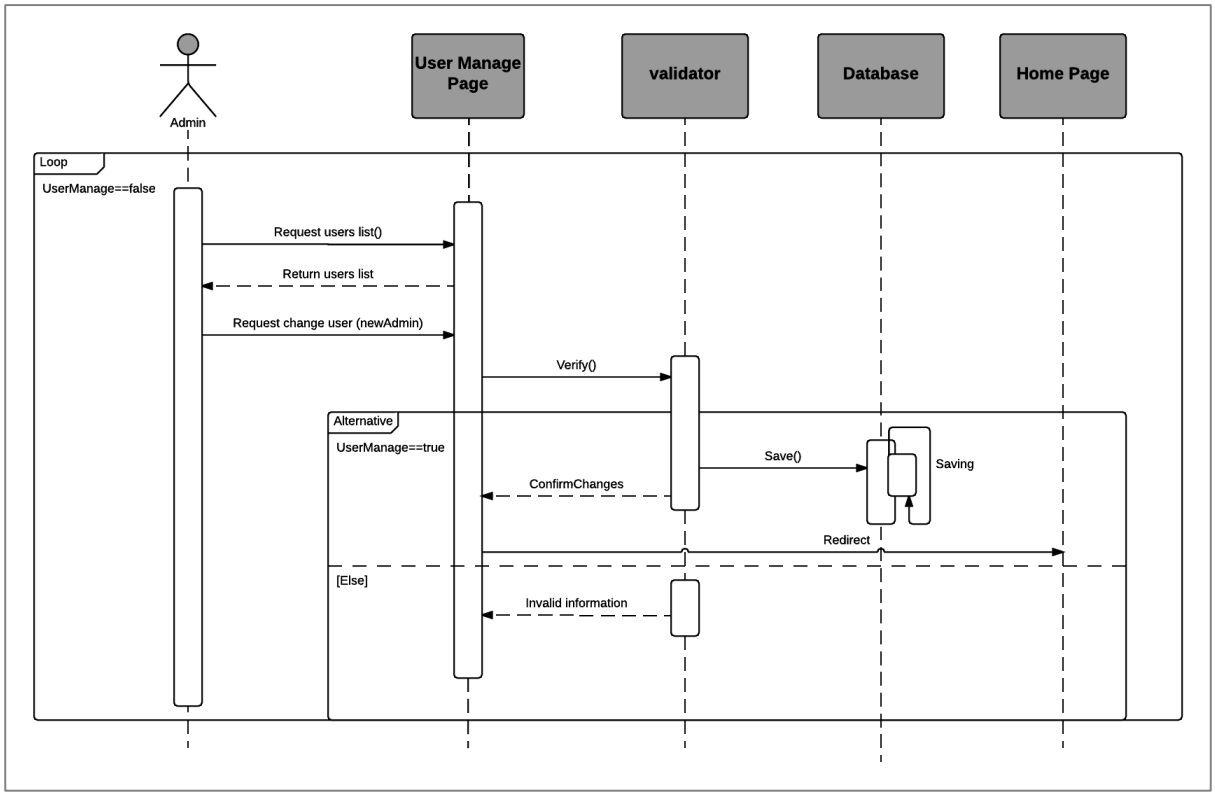
- **Manage System**



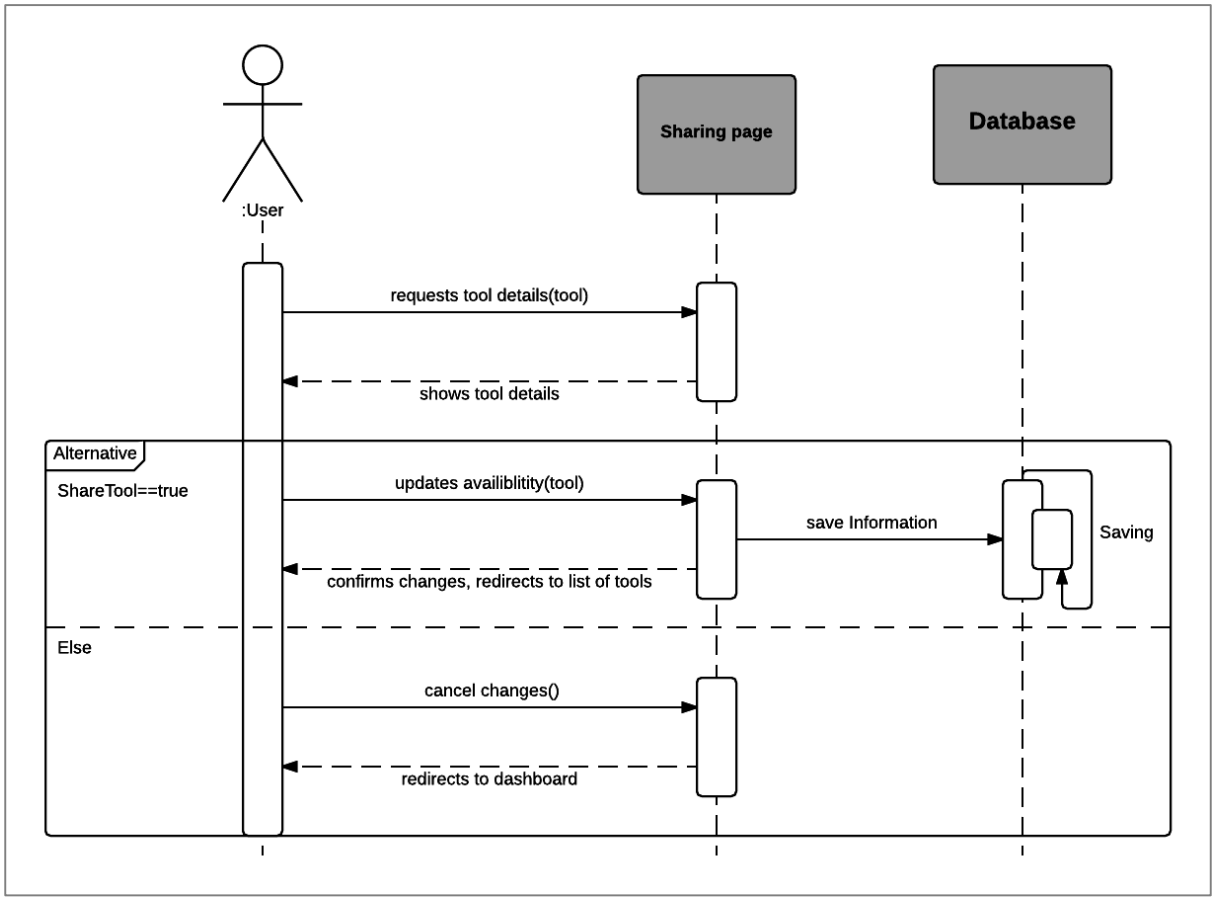
❖ **Community Statistic**



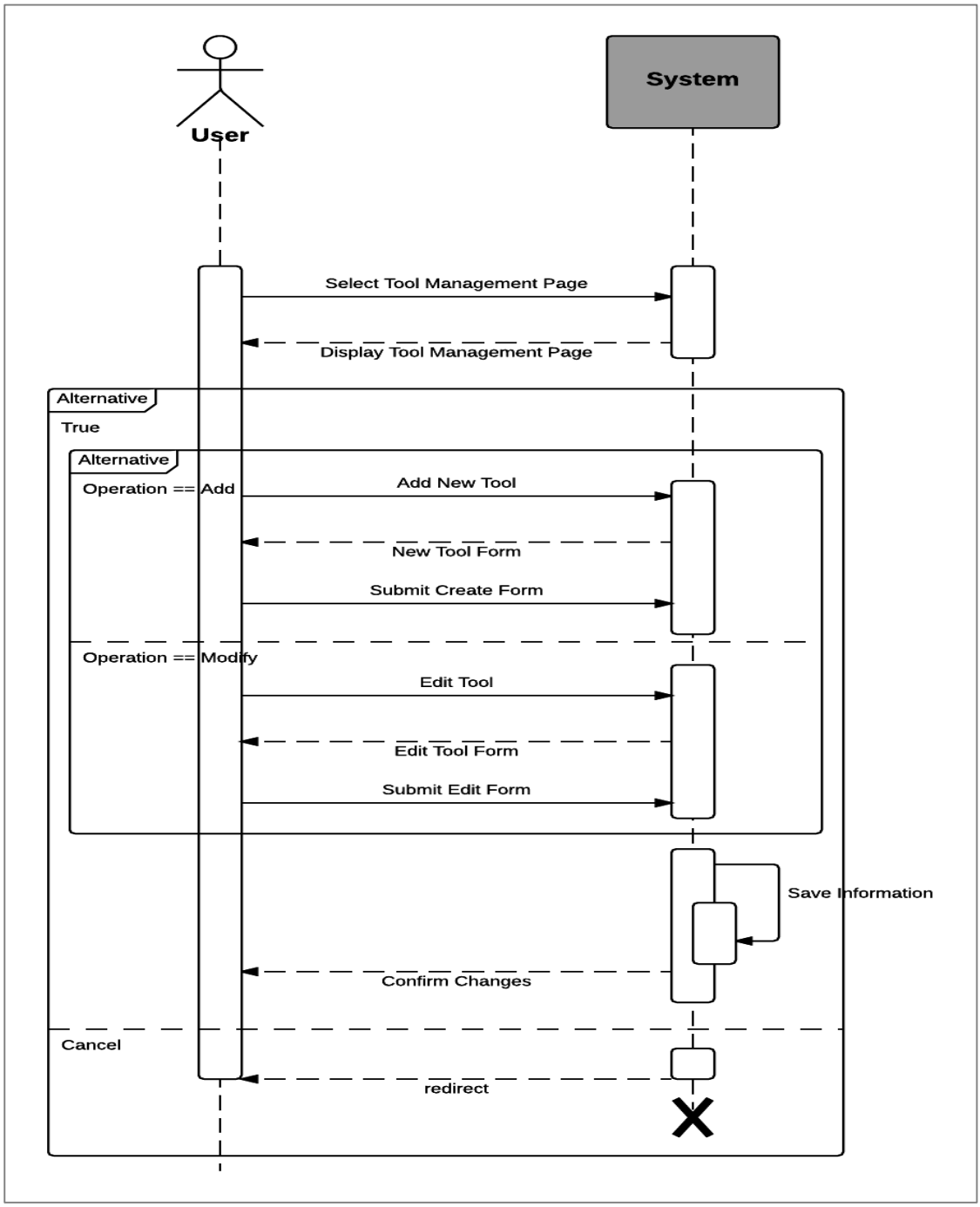
❖ User Manage


Admin
User Manage Page
validator
Database
Home Page
Loop
UserManage==false
Request users list()
Return users list
Request change user (newAdmin)
Verify()
Alternative
UserManage==true
Save()
Saving
ConfirmChanges
Redirect
[Else]
Invalid information

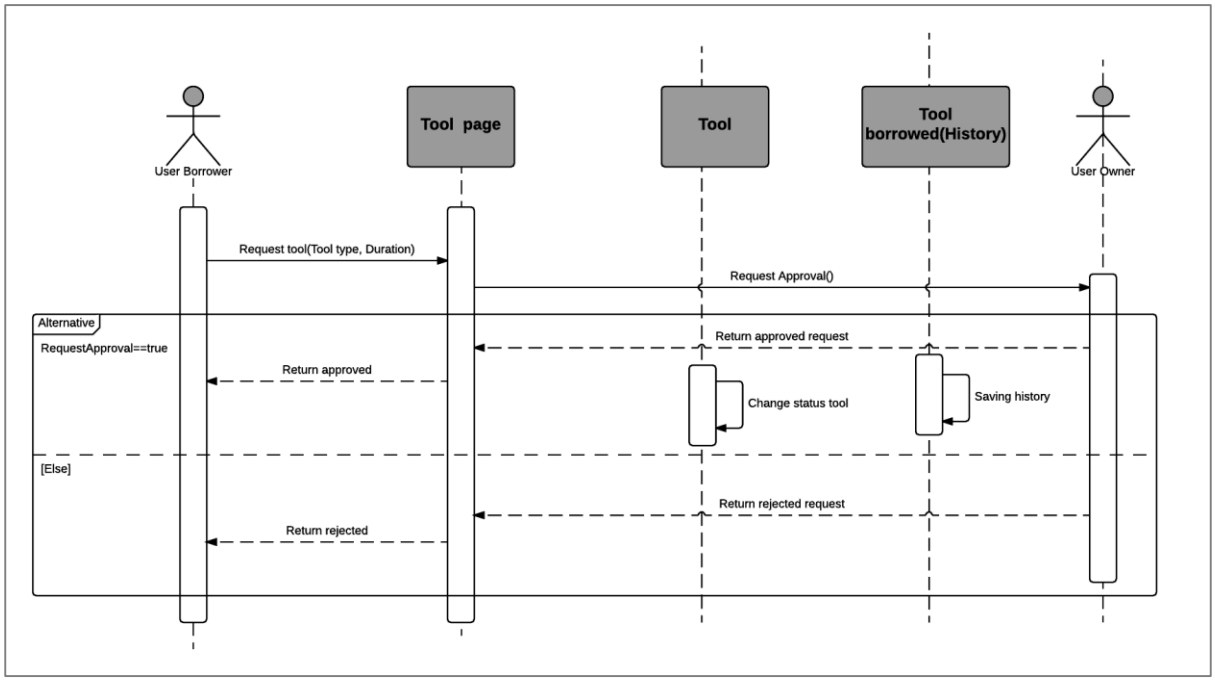
❖ **Share Tools**


:User
Sharing page
Database
requests tool details(tool)
shows tool details
Alternative
ShareTool==true
updates availiblitity(tool)
save Information
Saving
confirms changes, redirects to list of tools
Else
cancel changes()
redirects to dashboard

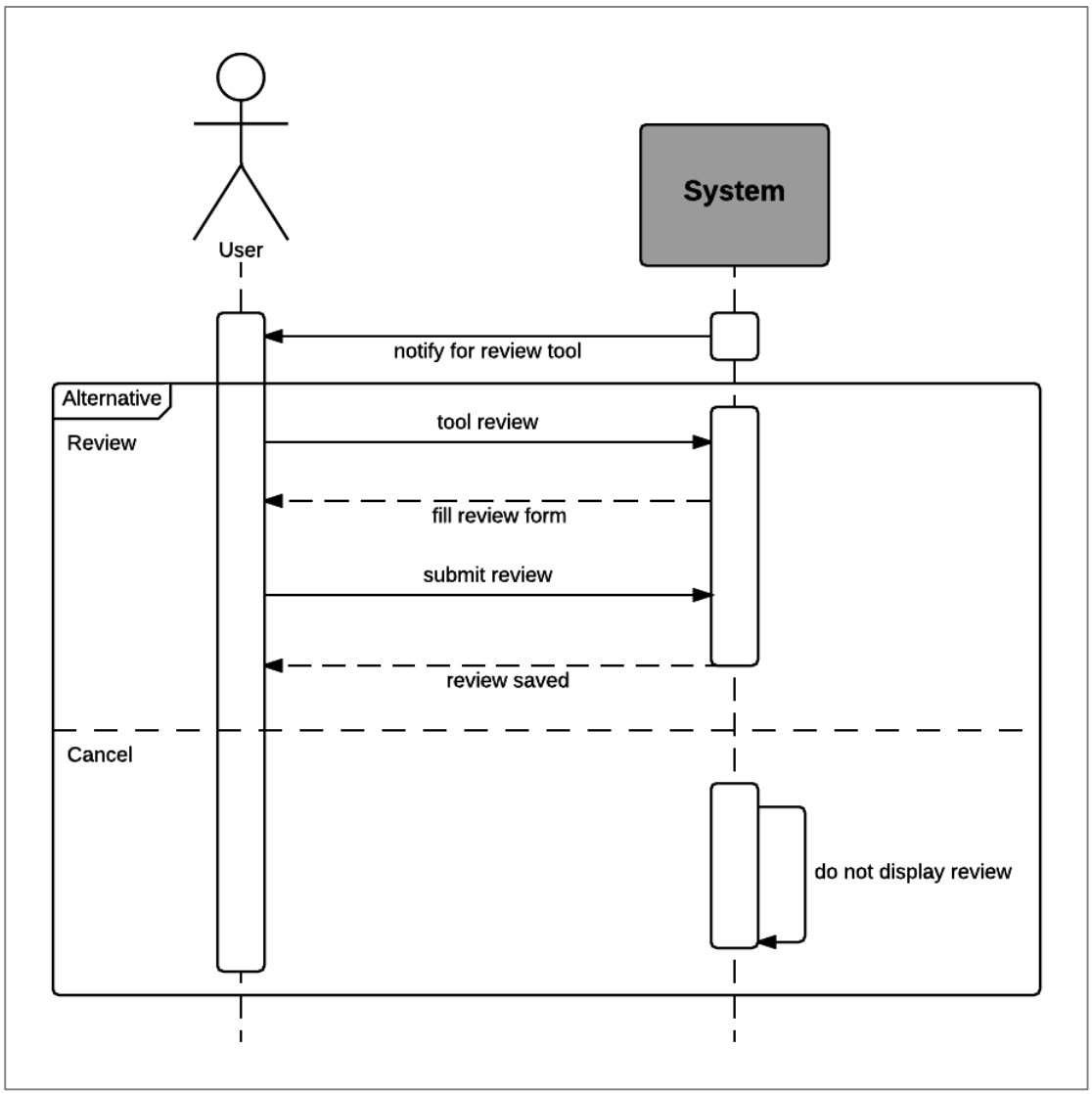
❖ **Manage Tool**


User
System
Select Tool Management Page
Display Tool Management Page
Alternative
True
Alternative
Operation == Add
Add New Tool
New Tool Form
Submit Create Form
Operation == Modify
Edit Tool
Edit Tool Form
Submit Edit Form
Save Information
Confirm Changes
Cancel
redirect

❖ Borrow Tools



❖ **Review Tools**



Design Rationale

We added a history table so we can keep track of the borrowing history. This will also help to show statistics to the website. That section will take the information about most borrowed tools, most active lenders, most actives borrowers, and most borrowed tools categories.

We separated user and admin in two different classes. Before the changes, we had a single user class with flag used to determine if the user had administrative privileges or not. However, the customer suggested to have a separated class called Admin that will inherit from the user class and it will have extra functionalities for administrators. This way if in the future we want to levels of administrators we they can inherit from user instead of modifying the user class. It will make it more reusable.

We decided to add a category class because this will help us to display statistics about tools for different types. The community can see if tools from a specific category are borrowed the most and they can have the chance to get more of these if they don't have enough.

We discussed with the customer and the idea of having a user with more than one phone number and more than one email was discarded. We also changed our design so pick up times will be related to tools instead of users. Tools can have more than one pick up time. Since an owner schedule may vary per week and even vary the hour availability per day.

We added a request class to keep track of all the requests and their status. This will also have the information about the last borrower of a tool so the owner knows who has his/her tools. The borrower can also check his/her requests and in case a request was rejected, they can see the reason given by the owner.

In addition, we removed the borrower id from the tool class, so now we will take that information from the history class. We discussed and concluded that the borrower id was already reflected in the requests class so we can search based on the tool id and the most recent date to get this information.

We suggested to the client to add review for the borrowed tools so that they can rate them. This would help other users to decide to borrow that tool that has a better condition.

Based on the customers' requirements, the payment is not added to the application. In the future this feature might be added to the application so that the payment could go as donation or personal use if preferred. Also the customer said that there should be one and only one shed for shared zone

According to the customers' requirement, the reviewing tool feature is mainly for the borrowers not for the owners. As the owner is providing the tool for free and would not mind if it is rated low or high.

The application would allow multiple requests to a specific tool. In the future if the owners could review the returned tool. For example, the returned tool is in good condition, so there might be a priority for the requests based on the status of the tool returned by that borrower.