Team: Riley Johnson, Vinnan Muralikrishnan
Travis Fox, Regina Van Driel

Step 1:

Strategies for gathering requirements:

Problem Analysis. Problem Analysis would be very helpful for this small news business. This strategy is a good one to use to solve current problems. Being a new site, it is important to figure out what sort of issues that users face, what kind of things they find issues with, and overall what on the site they find does not work. Ease of use is a big issue for websites and having users report problems is important.

Root Cause Analysis. Because *shortstop.news* is a new business, when users report issues with the site the initial problem could be a smaller issue or a symptom of a larger issue which could lead to the root cause of the problems. This analysis could be more challenging as well because finding the root problem is rarely easy.

Activity-Based Costing. This strategy would be used to examine the costs of the business and assign costs to them. In a startup, it is important to track where, when, and how the invested money is being used. Assigning costs to every part of the business, from materials to labor, can help to keep track of the total cost for the business.

Informal Benchmarking. *shortstop.news* is a new business based around news and using the benchmarking strategy could help to better understand what other news sites and outlets are doing right that could for this site as well. Informal benchmarking will help to find what could be added to the site that the people currently working on it could have not thought of before that could add more value to its users.

Outcome Analysis. Using the outcome analysis strategy, a systems analyst would encourage those working on the site to think in the perspective of the customer to better understand the value that the product. Understanding the perspective of the customer or user, can help the project manager to help enable the customers to do as much as they can do with the available product.

Technology Analysis. This strategy will ensure that the technology will stay relevant which would be important as this business is largely web-based. Allowing room for the website to be adaptable to changing technology can help it continue to be useful and meet the new goals for the organization.

Activity Elimination. As with all businesses over time, there may be certain activities that are still being performed but no longer required. Using the activity elimination strategy,

the business would test how the system would work when removing each activity in the process and assessing how the system would still perform.

Techniques for Requirements-Gathering:

Interview. The most common way to understand what the customers want is to ask them questions regarding the system. This technique works well because it is more of a one on one approach to finding out what the users really think about the system and any problems that they may run into.

Questionnaire. This technique could be used if there are already a few things that the developer team may think could be important or useful. This technique is similar to interviewing the customers only it has predetermined questions.

Observation. By using observation, information can be gathered by watching the processes being performed. This could also be useful to follow the way that some customers use the product in every way, seeing their issues happening first-hand.

Step 2:

Determining requirements for the system:

Non-functional requirements

1. Operational Requirements

1.1. The system will operate in Windows and Apple environment.

1.2. The system should automatically back up at the end of each day.

1.3 The back-end system should auto-update every time a new is published.

1.4 Update and implement the latest technologies in web development

1.5 Eliminate activities that are no longer required.

2. Performance Requirements

2.1. The system will load a page in 2 seconds or less.

2.2. The system will generally follow the 3-click rule.

3. Security Requirements

3.1. Secured payment option for premium users.

3.2. Only the system admin can decide which news feed to keep or remove.

4. Cultural and Political Requirements

4.1. Maintain cultural and political sensitivity as it covers international news.

5. Usability

5.1 User-friendly, even for users with very little computer knowledge.

5.2 Using outcome analysis should produce better news feed to users.

6. Availability

6.1 The system should be up and running 24/7.

7.Capacity

7.1 The server should be able to handle multiple user requests at the same time.

Functional Requirements:

1. Manage User reports

   1.1 Identify the root cause for the problem and fix it.

2. Manage Premium Users

   2.1 Maintain a record of all premium users and update it on a timely basis.

3. Audit Tracking

   3.1 Track all the costs from material to labor.

4. Authorization levels

   4.1 Set authorization for project manager, public relations, tech head, tech analyst, users and premium users.

5. Legal or Regulatory Requirements

   5.1 Comply with the latest legal requirements including copyrights.

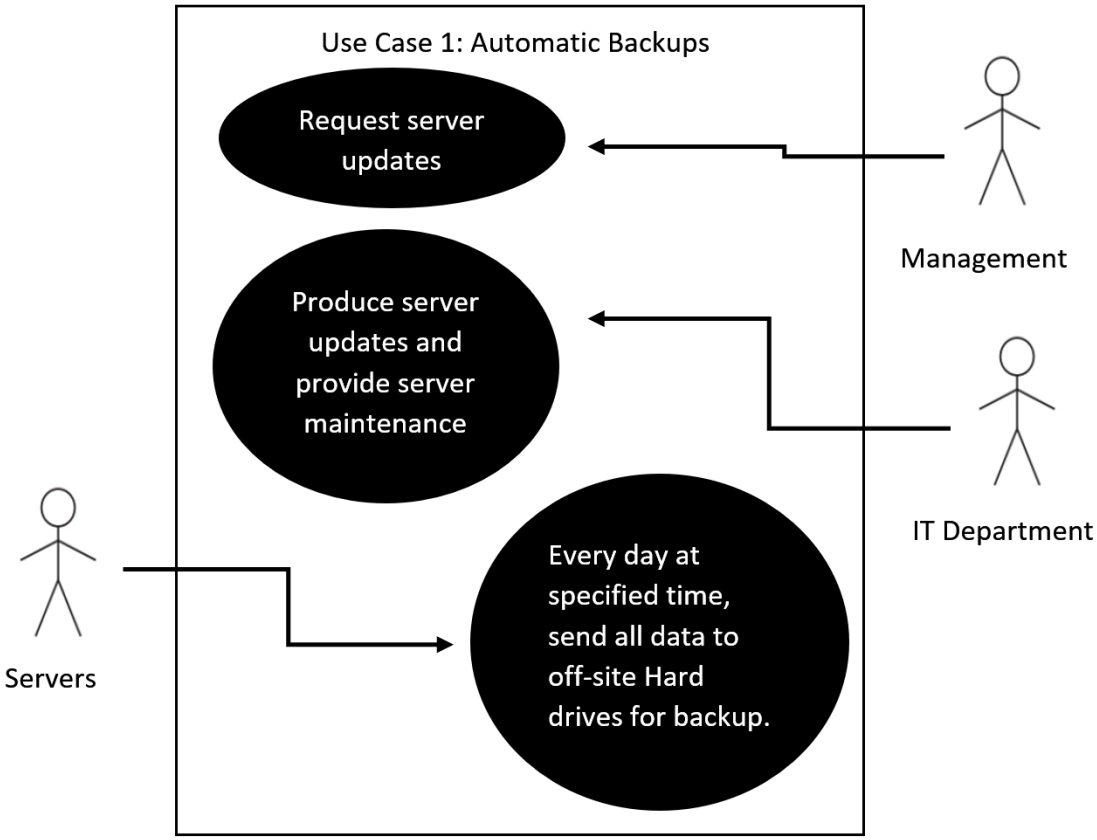
.

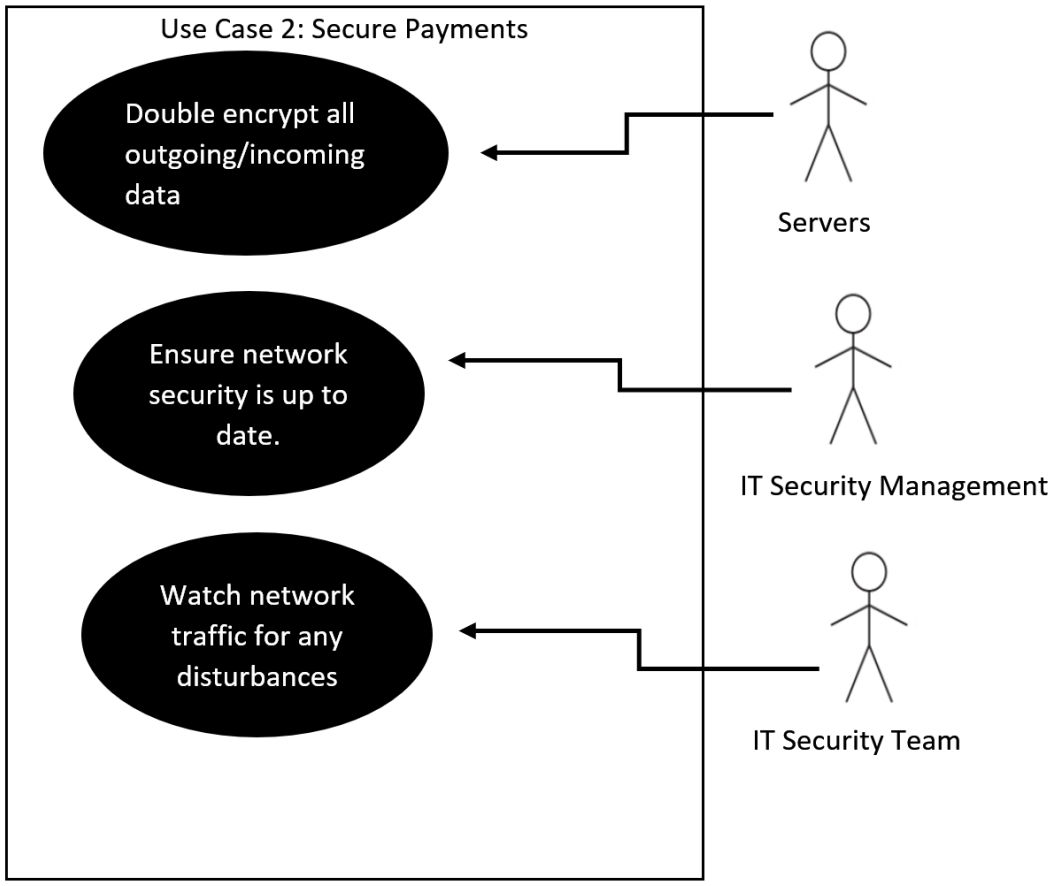
Step 3:

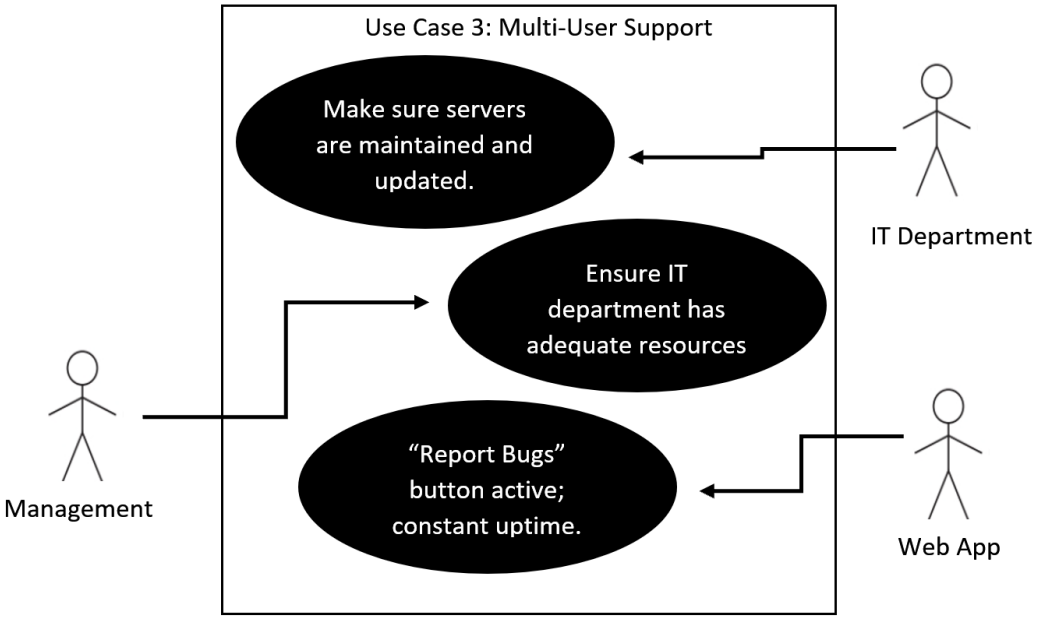
Use Cases:

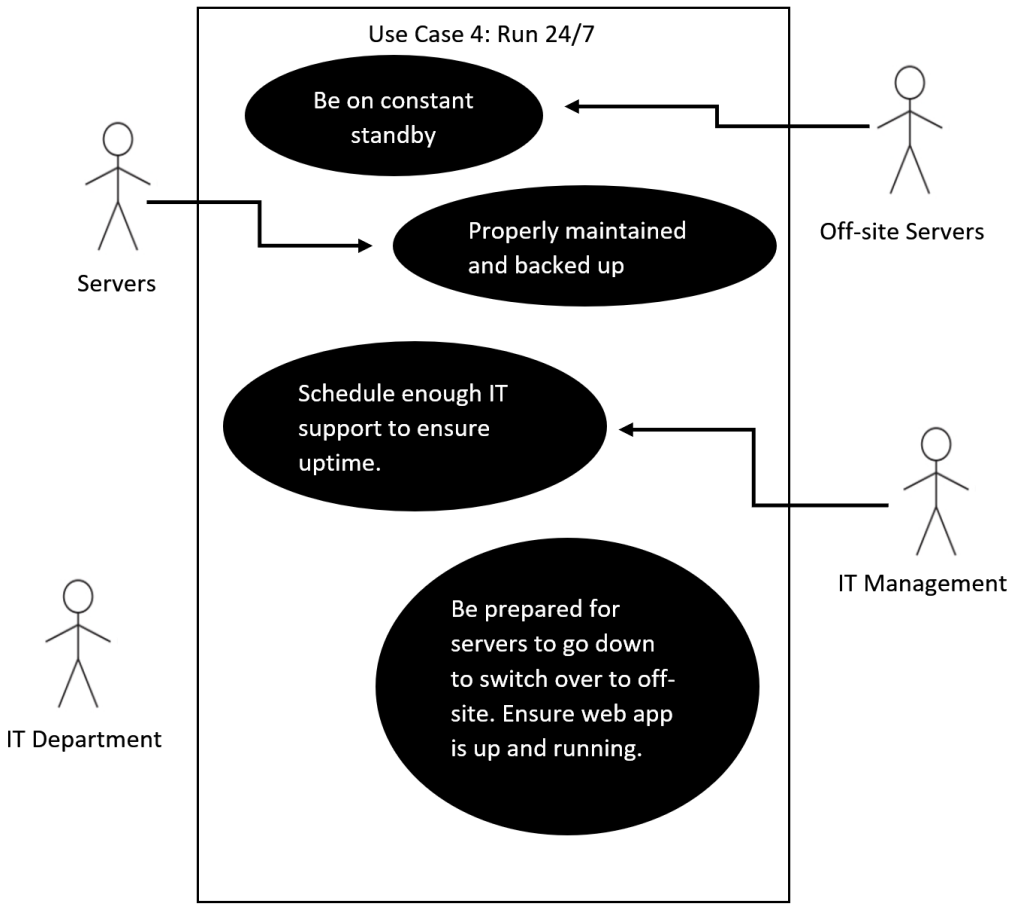
1. **Automatic Backups**: The system will have automatic backups
2. **Secure Payments**: Any payments will be made securely
3. **Multi-User Support**: There will be multi user support
4. **Run 24/7**: The system will be up and running 24/7
5. **Friendly Website:** Friendly to all users and skill levels

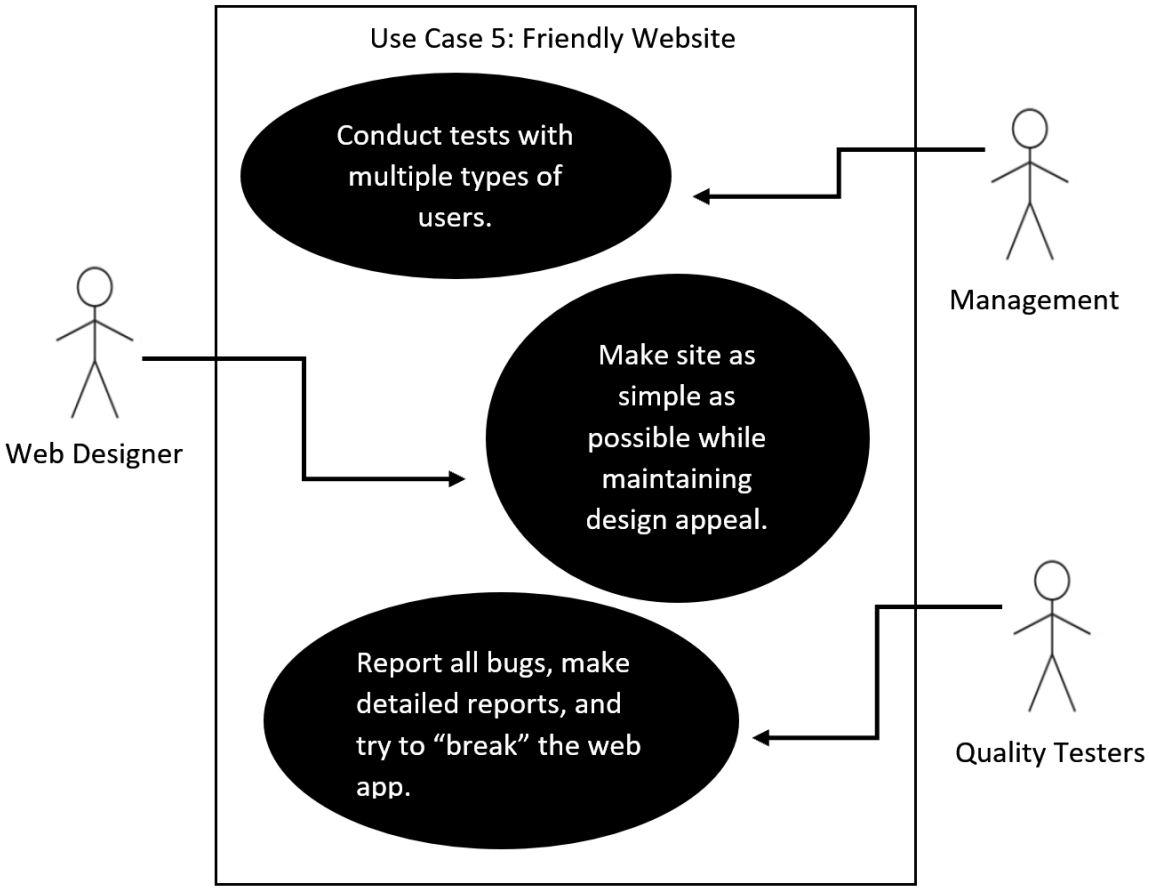
Diagrams:


Use Case 1: Automatic Backups
Request server updates
Produce server updates and provide server maintenance
Every day at specified time, send all data to off-site Hard drives for backup.
Management
IT Department
Servers


Use Case 2: Secure Payments
Double encrypt all outgoing/incoming data
Ensure network security is up to date.
Watch network traffic for any disturbances
Servers
IT Security Management
IT Security Team


Use Case 3: Multi-User Support
Make sure servers are maintained and updated.
Ensure IT department has adequate resources
"Report Bugs" button active; constant uptime.
IT Department
Management
Web App


Use Case 4: Run 24/7
Be on constant standby
Properly maintained and backed up
Schedule enough IT support to ensure uptime.
Be prepared for servers to go down to switch over to off-site. Ensure web app is up and running.
Servers
Off-site Servers
IT Management
IT Department


Use Case 5: Friendly Website
Conduct tests with multiple types of users.
Make site as simple as possible while maintaining design appeal.
Report all bugs, make detailed reports, and try to “break” the web app.
Management
Web Designer
Quality Testers

Descriptions:

Use Case Name: Automatic Backups	ID Number: 46
Short Description: The system makes automatic backups to Off-Site and 'In-shop" Hard-drives.	
Trigger: Type External / Temporal	
Major Inputs: Description: Source: Set Time IT Department	Major Outputs: Description: Source: Servers In Shop Drives Servers Off Site Drives
Major Steps Performed: 1. IT Department sets a time that the site is down for a short period of time in order to do the daily server backup. 2. When set time is 15 minutes away, website displays a banner notifying users that downtime backup approaches. 3. When set time happens, website shuts down and backup happens 4. Once backup is completed, website returns to service.	Information for Steps:

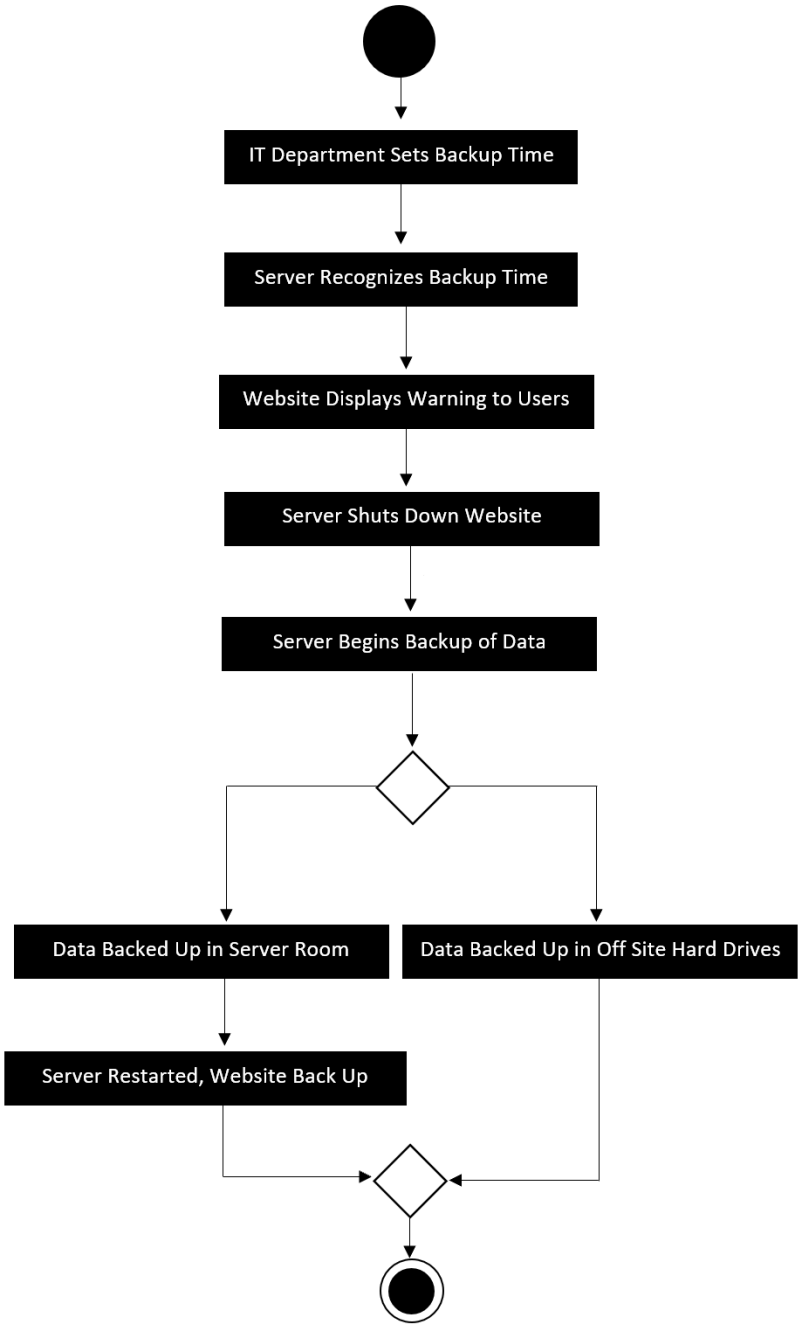
<table>
<tr><td>Use Case Name: Secure Payments</td><td>ID Number: 32</td></tr>
<tr><td colspan="2">Short Description: All payments will be made secure by using a hashed encryption as well as a shift encryption.</td></tr>
<tr><td colspan="2">Trigger: Type External / Temporal</td></tr>
<tr><td>Major Inputs:
Description: Source:
Payment Data Entered Customer</td><td>Major Outputs:
Description Source:
Data Encrypted Website
Data Double Encrypted Servers
Data Saved Servers</td></tr>
<tr><td>Major Steps Performed:
1. User inputs payment information.
2. Website encrypts data
3. Server encrypts data
4. Data is saved</td><td>Information for Steps:
When the user inputs they're payment information and clicks the submit button the web app encrypts the data before sending it to the server.

Once the data hits the server, it is again encrypted using a different hash before it is saved.</td></tr>
</table>

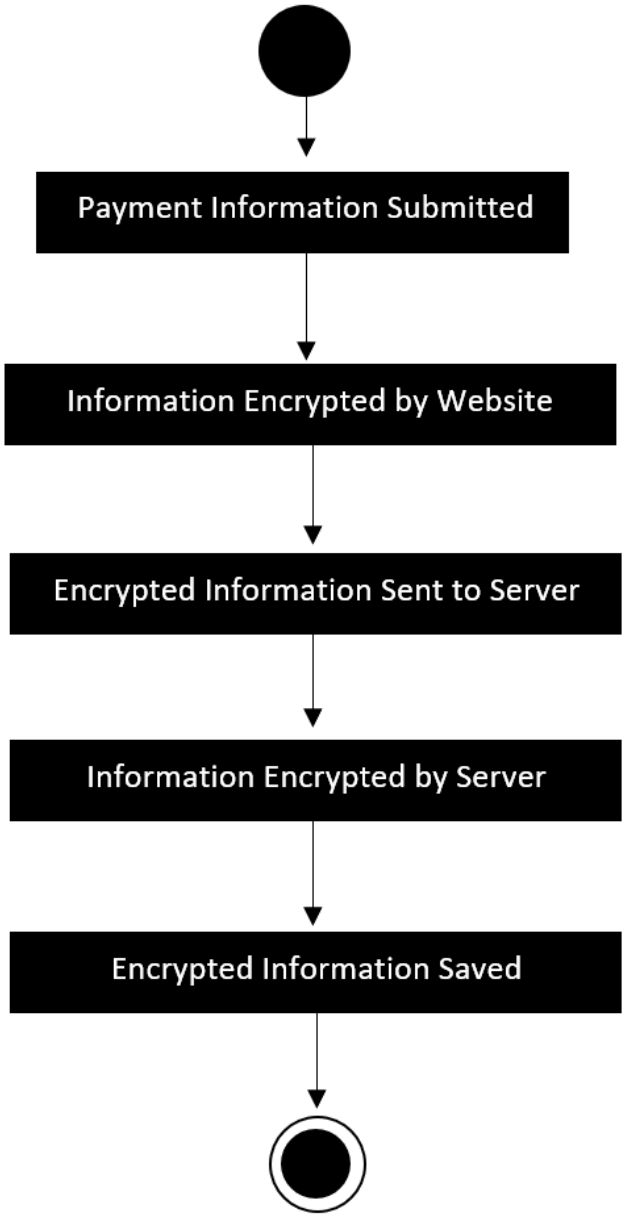
Step 4:

Activity Diagrams:

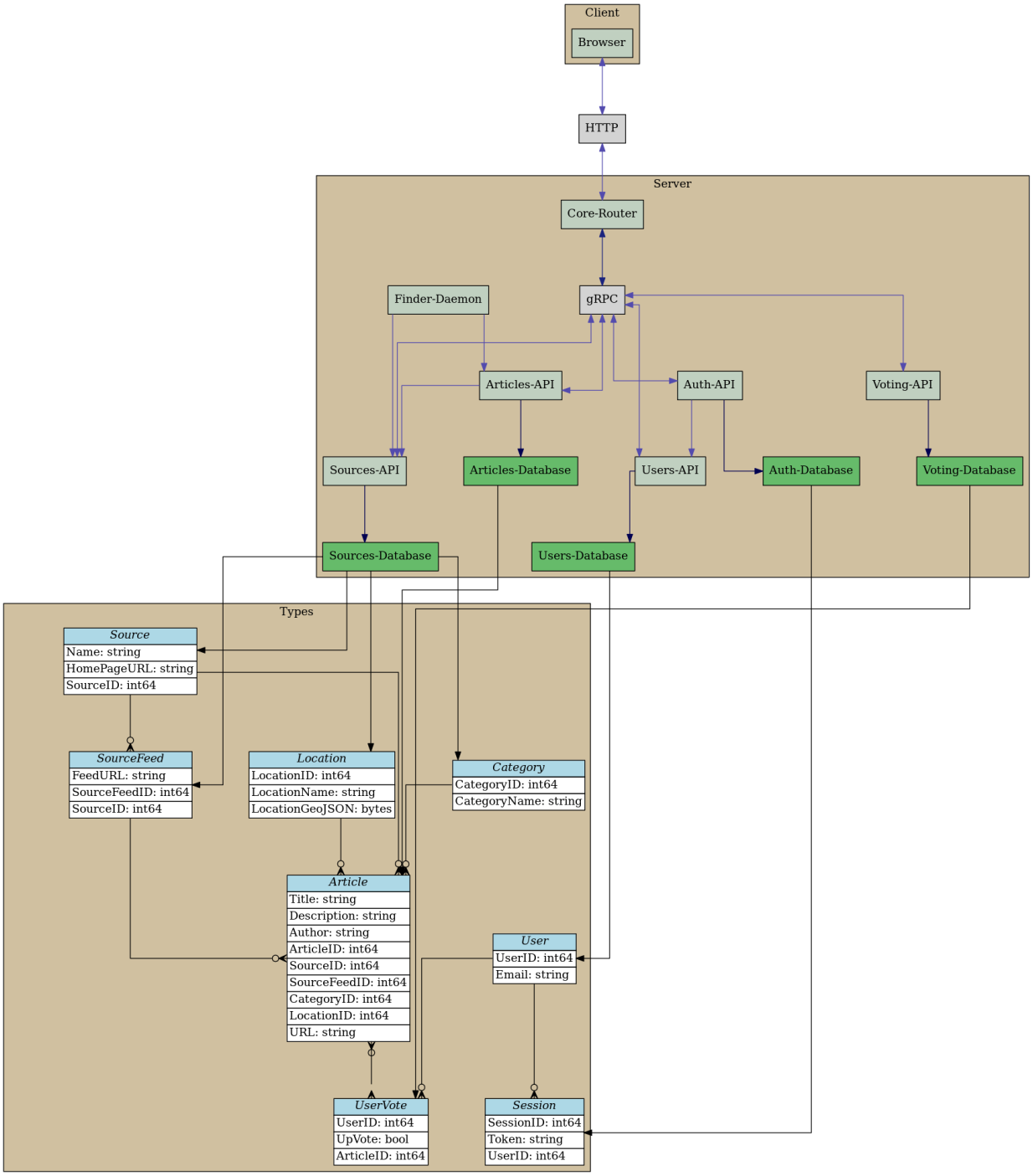
Use Case 1: Automatic Backups:



Use Case 2: Secure Payments:



Step 5:


Client
Browser
HTTP
Server
Core-Router
Finder-Daemon
gRPC
Articles-API
Auth-API
Voting-API
Sources-API
Articles-Database
Users-API
Auth-Database
Voting-Database
Sources-Database
Users-Database
Types
Source
Name: string
HomePageURL: string
SourceID: int64
SourceFeed
FeedURL: string
SourceFeedID: int64
SourceID: int64
Location
LocationID: int64
LocationName: string
LocationGeoJSON: bytes
Category
CategoryID: int64
CategoryName: string
Article
Title: string
Description: string
Author: string
ArticleID: int64
SourceID: int64
SourceFeedID: int64
CategoryID: int64
LocationID: int64
URL: string
User
UserID: int64
Email: string
UserVote
UserID: int64
UpVote: bool
ArticleID: int64
Session
SessionID: int64
Token: string
UserID: int64