

GetTrippy

Use-Case Model overview with Individual use case specifications

Abstract

This document provides an overview of the solution context, scope and high-level functional capabilities for the GetTripper project. It catalogs the actors that interact with the system and the use cases that together describe all the ways in which they interact with the system. It also documents the use-case diagrams that show the relationships that exist between the actors and the use cases.

Version History

Date	Version	Author	Description
7/13/2015	0.1	Dustin Owen	Edit Template
7/14/2015	0.2	DO & BH	Added Actors
7/16/2015	0.3	DO & BH	Added Use-Case catalog
7/17/2015	0.4	DO, BH & CF	Added Use-Cases
7/17/2015	0.5	DO, BH & CF	Added Diagram, Stories and touched up document.



An Essential Unified Process Document

Table of Contents

1	Introduction.....	4
1.1	Document Purpose	4
1.2	Document Scope	4
1.3	Document Overview	4
2	Use-Case Diagrams.....	5
2.1	<Diagram Name>.....	Error! Bookmark not defined.
3	Actor Catalog	5
4	Use-Case Catalog.....	6
5	Use case Specification x	8
5.1	Introduction	8
5.1.1	Document Purpose	8
5.1.2	Document Scope	8
5.1.3	Document Overview	8
5.2	Brief Description	10
5.3	Preconditions	10
5.4	Basic Flow.....	10
5.5	Alternative Flows	10
5.5.1	<Area of Functionality>.....	Error! Bookmark not defined.
5.5.2	<Area of Functionality>.....	Error! Bookmark not defined.
5.6	Post-Conditions	11
5.7	Special Requirements	12
5.8	Extensibility	12
5.9	Scenarios	12
5.10	Additional Information.....	Error! Bookmark not defined.

1 Introduction

1.1 Document Purpose

The purpose of this document is to provide an overview of the system use-case model in order to provide a high-level understanding of:

- Context – the people or things that interact with the system (the *Actors*)
- Scope – the things of value that the system performs for its Actors (the *Use Cases*).

1.2 Document Scope

The scope of this document is limited to:

- Diagramming and cataloging the system actors and use cases for the GitTrippy project.

The scope of this document does *not* include consideration of:

- Detailed specification of each use case – this is provided separately in a Use-Case Specification document for each use case.

1.3 Document Overview

This document contains the following sections:

- **Brief Description** – reminder of why the use case is needed
- **Use-Case Model Overview** – overview of the use-case model in the form of one or more use-case diagrams with supporting explanatory text
- **Actor Catalog** – catalog of all system actors
- **Use Case Catalog** – catalog of all system use-cases
- **References** – provides full reference details for all documents, white papers and books that are referenced by this document.

1.4 User Stories

This is the user stories for the use-cases.

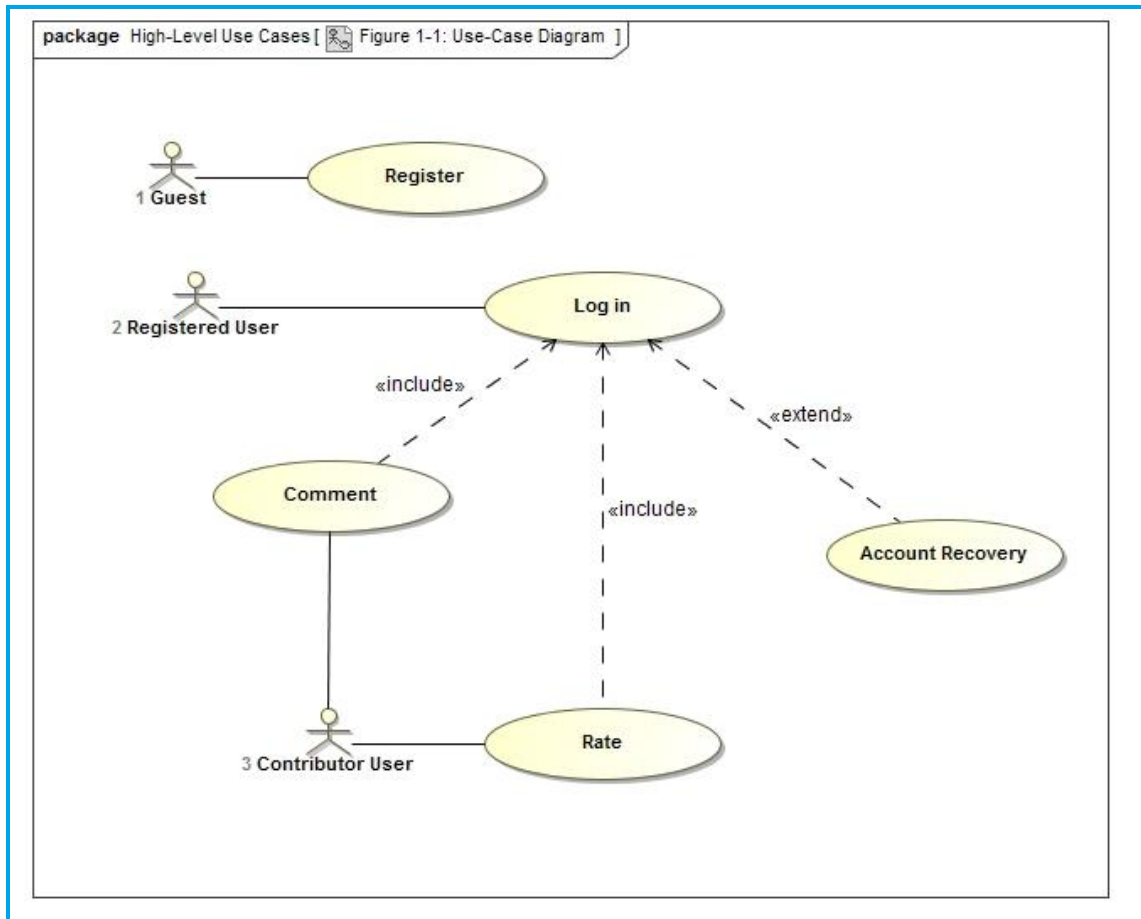
- I as a guest need register function to be able to achieve the goal of becoming a registered user
- I as a registered user need log in function to be able to achieve the goal of being able to rate and comment on website
- I as a contributor user need rate function to be able to achieve the goal of being able to contribute to the web site.

2 Use-Case Diagrams

This section provides an overview of the use-case model in the form of one or more use-case diagrams with supporting explanatory text.

2.1 Basic Interactions

This diagram shows the Actor/Use-Case relationship for the use-cases described in this document.



3 Actor Catalog

The table below catalogs the system actors, specifying for each actor:

- Name – unique and meaningful name for the actor
- Brief Description – summarizing the role that the actor plays with respect to the system.

Name	Brief Description
Guest	A user who hasn't registered with the site.
Registered User	A user who has registered but isn't logged on.

Name	Brief Description
Contributor User	A registered user who is logged on. Submits ratings and/or comments to items.
Content Reviewer	When someone submits new content, this person reviews them. Either approves or denies them.

4 Use-Case Catalog

The table below catalogs the system use cases, specifying for each use case:

- ID – unique identifier for the use case
- Name – unique and meaningful name for the use case
- Type:
 - *Base* – end-to end interaction between an actor and the system
 - *Inclusion* – abstracted common part of many Base Use Cases that is explicitly referenced by these use cases
 - *Extension* – extension of one or more referenced Base Use-Cases
 - *Abstract* – use case describing generic aspects of many Base Use Cases which in turn specialize this general case in some way.
- Brief Description – summarizes the use-case purpose in terms of the value produced for its actors and other stakeholders.

ID	Name	Type	Brief Description
1	Register	Base	Collects data from user to create an account.
2	Login	Base/Inclusion	Accepts a username and password to log the user into the system.
3	RateItem	Base	Accepts a rating value from a logged in user and submits the value into the database.
4	CommentItem	Base	Accepts a string value from a logged in user and submits the value into the database.
5	AccountRecovery	Extension	An extension class of Login which prompts the user of our account recovery options when a login attempt fails credentials.

ID	Name	Type	Brief Description

5 Use case Specification

One section per use cases – according to the use case specification template below

The Register Use-Case Specification specifies all requirements relating to this use case, including the flows through the use case and any non-functional requirements that relate specifically to this use case.

5.1 Introduction

5.1.1 Document Purpose

The primary objectives of the Use-Case Specification are to:

- Provide a complete set of requirements relating to the use case, including the flows through the use case and all significant external behavior of the use case
- Provide additional supporting information to support the communication of the purpose, requirements, constraints and flows through the use case.

5.1.2 Document Scope

The scope of this document is limited to consideration of:

- The specification this specific use case, including any non-functional requirements and constraints that relate specifically to this use case.

This scope of this document does not include consideration of:

- Requirements that relate specifically to any other use case – these are described in separate specifications for the other use cases
- Requirements that do not relate to any specific use case – these are described in a separate Supplementary Requirements document.

5.1.3 Document Overview

This document contains the following sections:

- **Brief Description** – focusing on the purpose of the use case
- **Preconditions** – the state the system must be in before the use case can be performed
- **Basic Flow** – what normally happens when the use case is performed
- **Alternative Flows** – unusual, optional or exceptional use case behavior

- **Post-Conditions** – valid system states after the use case has finished
- **Special Requirements** – non-functional requirements that relate specifically to this use case
- **Extensibility** – where the use case can be extended by other use cases
- **Scenarios** – lists the key representative scenarios for this use-case
- **Additional Information** – any other useful supporting information
- **References** – provides full reference details for all documents, white papers and books that are referenced by this document.

5.2 *Brief Description*

Collects data from user, validates the data and uses the data to create an account.

5.3 *Preconditions*

- The username and e-mail address must not already exist to another registered account.
- The password must contain only letters and digits, and a minimum of six characters.

5.4 *Basic Flow*

{Clicks Register}

1. The use-case scenario begins when the Guest clicks register.
2. The system loads the registration form.

{Complete fields}

{Clicks submit}

3. The system validates the username and e-mail address doesn't already exist to another registered account.
4. System verifies that the password is at least 6 characters long and contains only letters and digits.
5. System takes user's info and creates the corresponding fields in the database.
6. Displays screen indicating the registration process was successful.

5.5 *Alternative Flows*

This section describes the alternative flows for the use case, grouped into named subsections reflecting the area of functionality to which the alternative flow relates. Each alternative flow is described in terms of:

- The point in the use case at which the alternative flow might begin
- The conditions under which the alternative flow will begin
- The flow of events within the alternative flow
- The point at which the use case resumes on completion of the alternative flow.

5.5.1 *Incorrect User Info*

5.5.1.1 Handle the scenario when a user inputs invalid data.

At {Clicks submit} if the customer's information fails the validation by the system:

1. The system will advise the customer of the error(s) that exist in their input.

2. The customer enters in new information.
3. Resume at {Clicks submit}

5.6 Post-Conditions

- The system has rejected the user's input, and the customer hasn't corrected it and remains unregistered.
- The system has accepted the user's input, and the customer is now a Registered User.

5.7 Special Requirements

The system must enforce the input validation criteria of having no more than once occurrence of an e-mail address, or username in the database.

The system must enforce the input validation criteria of having the password be a minimum of 6 characters and contain only letters and numbers.

5.8 Extensibility

No public extension points defined.

5.9 Scenarios

A scenario is an instance or specific occurrence of a use case. This section lists key representative scenarios for this use-case, listing for each scenario:

- Name – a unique and meaningful name for the scenario
- Flows Exercised – List of the use case flows exercised by the scenario
- Additional Notes – any supporting notes about the scenario or its purpose.

Name	Flows Exercised	Additional Notes
Quick Registration	Basic Flow	User visits site and wants to join.
Problematic Registration	Incorrect User Info	User visits site and wants to join, but enters registration information that doesn't meet our system requirements.

6 Use case Specification

One section per use cases – according to the use case specification template below

The Log In Use-Case Specification specifies all requirements relating to this use case, including the flows through the use case and any non-functional requirements that relate specifically to this use case.

6.1 Introduction

6.1.1 Document Purpose

The primary objectives of the Use-Case Specification are to:

- Provide a complete set of requirements relating to the use case, including the flows through the use case and all significant external behavior of the use case
- Provide additional supporting information to support the communication of the purpose, requirements, constraints and flows through the use case.

6.1.2 Document Scope

The scope of this document is limited to consideration of:

- The specification this specific use case, including any non-functional requirements and constraints that relate specifically to this use case.

This scope of this document does not include consideration of:

- Requirements that relate specifically to any other use case – these are described in separate specifications for the other use cases
- Requirements that do not relate to any specific use case – these are described in a separate Supplementary Requirements document.

6.1.3 Document Overview

This document contains the following sections:

- **Brief Description** – focusing on the purpose of the use case
- **Preconditions** – the state the system must be in before the use case can be performed
- **Basic Flow** – what normally happens when the use case is performed
- **Alternative Flows** – unusual, optional or exceptional use case behavior

- **Post-Conditions** – valid system states after the use case has finished
- **Special Requirements** – non-functional requirements that relate specifically to this use case
- **Extensibility** – where the use case can be extended by other use cases
- **Scenarios** – lists the key representative scenarios for this use-case
- **Additional Information** – any other useful supporting information
- **References** – provides full reference details for all documents, white papers and books that are referenced by this document.

6.2 Brief Description

Give the user access to the logged in status. This gives users access to features such as rate and comment.

6.3 Preconditions

User is already registered to the website.

6.4 Basic Flow

This section describes the normal flow through the use case in terms of what the actor does and what the system does in response, structured as a dialog between the actor and the system.

1. The registered user selects log in option on web site system
 - a. System displays log in window
2. The registered user enters username and password into text fields
 - a. System validates if user name and password is correct
3. If user name and password is not correct
 - a. System tells user to enter information again
 - b. If registered user does not know username or password user clicks on
4. If user name and password is correct then system displays main window for the registered user.

6.5 Alternative Flows

This section describes the alternative flows for the use case, grouped into named subsections reflecting the area of functionality to which the alternative flow relates. Each alternative flow is described in terms of:

- The point in the use case at which the alternative flow might begin
- The conditions under which the alternative flow will begin
- The flow of events within the alternative flow
- The point at which the use case resumes on completion of the alternative flow.

6.5.1 Account Recovery

6.5.1.1 {Registered user clicks on forgot username/password}

1. System displays username/password recovery window
2. Registered user fills out answers of security questions
 - a. System validates if user information is correct
3. If Registered user selects password recovery
 - a. System asks user to enter new password

- b. User enters new password
 - c. System validates if password meets all criteria
- 4. If password meets all criteria
 - a. System returns to log in screen
- 5. If Registered user selects username recovery
 - a. System displays registered user's username

6.6 Post-Conditions

- System has entered the main registered user screen
- System has stored new password if registered user restarted password

6.7 Special Requirements

- The system must validate the user's name to the password 100% of the time.

6.8 Extensibility

- Account Recovery

6.9 Scenarios

A scenario is an instance or specific occurrence of a use case. This section lists key representative scenarios for this use-case, listing for each scenario:

- Name – a unique and meaningful name for the scenario
- Flows Exercised – List of the use case flows exercised by the scenario
- Additional Notes – any supporting notes about the scenario or its purpose.

Name	Flows Exercised	Additional Notes
User Login	Basic Flow	User puts in a valid username and password and is logged into the system.
Failed Login	Account Recovery	User puts in invalid username and password and is provided an option to recover account credentials.

7 Use case Specification

One section per use cases – according to the use case specification template below

The RateItem Use-Case Specification specifies all requirements relating to this use case, including the flows through the use case and any non-functional requirements that relate specifically to this use case.

7.1 Introduction

7.1.1 Document Purpose

The primary objectives of the Use-Case Specification are to:

- Provide a complete set of requirements relating to the use case, including the flows through the use case and all significant external behavior of the use case
- Provide additional supporting information to support the communication of the purpose, requirements, constraints and flows through the use case.

7.1.2 Document Scope

The scope of this document is limited to consideration of:

- The specification this specific use case, including any non-functional requirements and constraints that relate specifically to this use case.

This scope of this document does not include consideration of:

- Requirements that relate specifically to any other use case – these are described in separate specifications for the other use cases
- Requirements that do not relate to any specific use case – these are described in a separate Supplementary Requirements document.

7.1.3 Document Overview

This document contains the following sections:

- **Brief Description** – focusing on the purpose of the use case
- **Preconditions** – the state the system must be in before the use case can performed
- **Basic Flow** – what normally happens when the use case is performed
- **Alternative Flows** – unusual, optional or exceptional use case behavior

- **Post-Conditions** – valid system states after the use case has finished
- **Special Requirements** – non-functional requirements that relate specifically to this use case
- **Extensibility** – where the use case can be extended by other use cases
- **Scenarios** – lists the key representative scenarios for this use-case
- **Additional Information** – any other useful supporting information
- **References** – provides full reference details for all documents, white papers and books that are referenced by this document.

7.2 **Brief Description**

Retrieves the user's rating for a particular item and stores it into the database.

7.3 **Preconditions**

User is already registered to the website.

7.4 **Basic Flow**

{Selects Rating}

{Clicks Submit}

1. System checks to determine if the user is logged in.
2. System retrieves the rating the user selected and stores it in the database for the particular item along with the user's info.

7.5 **Alternative Flows**

This section describes the alternative flows for the use case, grouped into named subsections reflecting the area of functionality to which the alternative flow relates. Each alternative flow is described in terms of:

- The point in the use case at which the alternative flow might begin
- The conditions under which the alternative flow will begin
- The flow of events within the alternative flow
- The point at which the use case resumes on completion of the alternative flow.

7.5.1 **Rate While Logged Out**

7.5.1.1 Handle the scenario when a user submits a rating before they have logged into the system.

At {Clicks Submit} if the customer isn't logged into the system:

1. The system will advise the customer they need to log in to use this feature.
2. Includes **LogIn**
3. If user logged then resume at {Clicks Submit}

7.6 **Post-Conditions**

- The user is logged in and the rating is successfully submitted.
- The user isn't logged in, but does so when prompted to and then the rating is successfully submitted.
- The user isn't logged in, and doesn't do so when prompted and the rating is disregarded.

7.7 Special Requirements

- The system must store the rating with the user's information 99% of the time.
- The system must ensure the user is logged in, or logs in when prompted in order to store the rating.

7.8 Extensibility

No public extension points defined.

7.9 Scenarios

A scenario is an instance or specific occurrence of a use case. This section lists key representative scenarios for this use-case, listing for each scenario:

- Name – a unique and meaningful name for the scenario
- Flows Exercised – List of the use case flows exercised by the scenario
- Additional Notes – any supporting notes about the scenario or its purpose.

Name	Flows Exercised	Additional Notes
Submits Rating	Basic Flow	User submits a rating for an item and is logged in.
Submits Rating Logged Out	Rate While Logged Out	User submits a rating to an item, but either didn't sign in, or has been logged out due to other reasons.