

RESUMASTER

Ashley Lu Couch

Brooklynn Stone

Jacob Curtis

Matthew McCaskill

Micah Schiewe

TABLE OF CONTENTS

<i>Abstract</i>	3
<i>Background</i>	4
<i>Glossary</i>	5
<i>Analysis</i>	7
Requirements	8
Use Cases.....	10
Traceability Matrix	18
Domain Model.....	20
System Sequence Diagrams.....	22
<i>Design</i>	38
Design Class Diagram	39
Package Diagram	41
Sequence Diagrams	43
GRASP	59
<i>Implementation</i>	61
Design Patterns	62
Online Resources.....	65
<i>GUI</i>	67
<i>Evaluation/Testing</i>	69
Static Code Analysis	71
Gantt Diagram	73
<i>Conclusion</i>	75
<i>Time Sheets</i>	76

ABSTRACT

This document describes the analysis, design, and implementation of the ResuMaster software. With the original intention of having a user system, some features have been voided, such as having a log-in process, a reviewer account, and the ability to leave comments on a resumé.

In the analysis portion of this document are the use cases, system sequence diagrams, and the domain model. Since the vision of the software had changed early on, some of these artifacts are no longer up to date, so a status update has been included for the use cases to indicated changes, voided processes, and unchanged processes. However, all diagrams are included for viewing purposes.

In the design portion of this document are the General Responsibility Assignment Software Patterns, design model, and various diagrams. The GRASP portion details which patterns are used in the software and where they can be found. More detail on these patterns can be found on the software website in the “Iteration 2” pdf. Diagrams detailing the implementation and structure of the software are also included in this section.

Finally, in the implementation portion of this document are the various design patterns being utilized, where they can be found, and why they were chosen in those instances. This portion also included a link to the git repository where the entirety of the code for the software can be viewed and downloaded. Also included is a link to a website detailing the background for this software and information regarding the development of ResuMaster.

BACKGROUND

Creating a custom resumé for each company is a pain. We plan to deliver a unified resumé software. This software allows users to store data they would spread across separate resumés. It allows the user to vary their resumés depending on the company they would be applying to. Resumé writers can export resumés containing specific data segments in a formatted PDF file. Users can request to have their resumé reviewed in part or whole without having to make an appointment. Reviewers can communicate their feedback in the form of email communication with the resumé writer. The user can then make edits to their resumé based on this feedback. The user will also be able to import and export their resumé data to ease writing their resumé across computers. This will benefit our users by reducing stress around creating a resumé and help them to apply to positions in a more efficient manner.

GLOSSARY

Resumé Terms:

- Category: An overarching section of the resumé. In a standard resumé, categories include work experience, education, biography, *et. cetera*.
- Experience Category: A type of Category which contains Experiences.
- Bullet Category: A type of Category which only contains Bullets.
- Text Category: A type of Category which contains a block of text. Used for profile or informational sections.
- Header Category: A type of Category containing typical header fields such as a name, address, and a link.
- Experience: A specific option for what goes into an Experience Category. This type of Resumé Item can contain Bullets.
- Bullet: A bullet point in an Experience or Bullet Category which allows the user to add text and display it in the Resumé.

System Terms:

- User: An individual recorded in the System as one who utilizes the service to write resumés. All Users have an email by which they can use to send their Resumé from.

File Terms:

- Resumé Data: A possible Category, Item, or Field written by a user in JSON format.

ANALYSIS

REQUIREMENTS

R01 Modify Resumé Fields

Description: A writer must be able to add, remove, and edit fields that are contained in categories within their Resumé. Here they will fill out their relevant information for the category they are writing in.

R02 Have Custom Categories

Description: As a Resumé writer, the user should be able to create custom Categories which they can add custom Items and fields to.

R05 Export Resumé

Description: Resumé writers must be able to export their Resumé in the form of a formatted PDF, and also as a data file to be used on another machine with the application installed - allowing users to work across multiple machines.

R06 Send Review Request

Description: As a writer, the user must be able to request a review for their resumé. This will be in the form of an email sent from the System/Writer to a reviewer with a request to review and leave feedback on the desired Resumé.

R08 Toggle Resumé Parts

Description: For the purpose of customizing Resumés for various companies and requesting reviews, Resumé writers should have the ability to toggle which categories should be included in their exported Resumés.

R09 Export the Resumé to a server

Description: As a writer, the user must be able to export their resumé to a server when they request a review. This will also ensure that the resumé is only accessible to a reviewer for a certain period of time.

USE CASES

UC01 - Create an account

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to create a new account that will let them log in and be able to make a resume.
- Admin: Wishes to allow users to have an account that is not a duplicate.

Preconditions: none

Postcondition: An account has been created for that user and they can now login normally

Status: Voided.

UC02 - Request Review

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to reliably notify a specific reviewer that his resumé is available to review.
- Reviewer: Wishes to be quickly notified of a request to review a resumé.
- Admin: Wishes to ensure non-Reviewers are not contacted regarding a resumé review.

Preconditions: The Writer is signed into the system.

Postcondition: The Reviewer has received an email notification that the Writer has a resumé to review. Alternatively, the Writer has cancelled the operation.

Status: No user system. Writer sends Resumé through email and receives feedback by email.

UC03 - Write Review

Scope: Resumé Review Software

Level: User goal

Primary Actor: Reviewer

Stakeholders and Interests:

- Reviewer: wants to leave a review on a resumé for the Writer
- Admin: wants the Reviewer to successfully leave feedback on a resumé for the Writer to see these comments when they view the review
- Writer: wants to have their resumé reviewed by a Reviewer

Preconditions: The Reviewer is signed into the System and has either received a request to review a resumé or is accessing a resumé to review

Postcondition: The Reviewer has left their review on the resumé and the Writer has been notified by the System

Status: Voided.

UC04i – Edit Item(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to edit a specific item and be able to save the change.
- Reviewer: Wishes to be able to see the modified value of the field.

Preconditions: The Writer knows which item(s) they would like to edit. Application is open to login page.

Postcondition: The item(s) that the Writer wanted to update have been changed and saved.

Status: Each Category type can be edited in a different way (Experience, Bullet, Text, or Header).

UC04f – Edit Field(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to edit a specific field and be able to save the change.
- Reviewer: Wishes to be able to see the modified value of the field.

Preconditions: The Writer knows which field(s) they would like to edit. Application is open to login page.

Postcondition: The field(s) that the Writer wanted to update have been changed and saved.

Status: Fields are specified within each Category type (Experience, Bullet, Text, or Header)

UC05a - Create category

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to create a new category that they can add resume elements to.

Preconditions: The Writer is signed into the system.

Postcondition: A category has been added or removed from the list of resume Categories.

Status: Category is pre-filled with information which can be changed at any time

UC05b - Remove category

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to remove a category that they no longer want to be part of a resume.

Preconditions: The Writer is signed into the system and has an available category to remove.

Postcondition: A category has been removed from the list of resume categories.

Status: Unchanged.

UC06 - Export Resumé

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to generate his resumé.
- Admin: Wishes to avoid cluttering the filesystem with many exported Writer resumés.

Preconditions: The Writer is signed into the system.

Postcondition: The User has received the final resumé.

Status: A Resumé can be exported with no active fields and no longer gives the user a preview before the export

UC08 - Check Review Comments

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: wants to read a Review on their resumé made by a Reviewer
- Admin: wants the Writer to successfully view feedback from a Reviewer
- Reviewer: wants the Writer to receive their feedback and be able to make constructive changes

Preconditions: The Writer is signed into the System and has received a notification that they have a Review to read or is checking for any past or new Reviews

Postcondition: The Writer has successfully checked for reviews or has read a review from a Reviewer

Status: Voided.

UC09ci – Create Item(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to create specific item(s) and to be able to save the change(s).
- Reviewer: Wishes to be able to see the modified version of the resume.

Preconditions: The Writer knows which item(s) they would like to create. Application is open to login page.

Postcondition: The items(s) that the Writer wanted to create have been changed and saved.

Status: This action has been turned into adding Experiences to an Experience Category

UC09ri – Remove Item(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to remove specific item(s) and to be able to save the change(s).
- Reviewer: Wishes to be able to see the modified version of the resume.

Preconditions: The Writer knows which item(s) they would like to remove. Application is open to login page.

Postcondition: The items(s) that the Writer wanted to remove have been changed and saved.

Status: This action has been turned into removing Experiences from an Experience Category.

UC09cf – Create Field(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to create specific fields(s) and to be able to save the change(s).
- Reviewer: Wishes to be able to see the modified version of the resume.

Preconditions: The Writer knows which fields(s) they would like to create. Application is open to login page.

Postcondition: The items(s) that the Writer wanted to create have been changed and saved.

Status: This action has been turned into adding Bullets to a Bullet Category or an Experience.

UC09rf – Remove Field(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to remove specific field(s) and to be able to save the change(s).
- Reviewer: Wishes to be able to see the modified version of the resume.

Preconditions: The Writer knows which field(s) they would like to remove. Application is open to login page.

Postcondition: The field(s) that the Writer wanted to remove have been changed and saved.

Status: This action has been turned into removing Bullets from a Bullet Category or an Experience

UC010 - Edit category

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to edit a category, name, type, or add an element

Preconditions: The Writer is signed into the system and a category already exists.

Postcondition: The information is saved to be referenced later.

Status: Unchanged.

UC11i – Toggle Item(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to toggle the visibility of specific item(s) and be able to save the change(s).
- Reviewer: Wishes to be able to see the modified toggles.

Preconditions: The Writer knows which item(s) they would like to toggle. Application is open to login page.

Postcondition: The item(s) that the Writer wanted to toggle have been toggled and saved.

Status: Once a Category has been toggled, the system must save the changes before toggling another Category

UC11f – Toggle Fields(s)

Scope: Resumé Review Software

Level: user-goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to toggle the visibility of specific field(s) and be able to save the change(s).
- Reviewer: Wishes to be able to see the modified toggles.

Preconditions: The Writer knows which field(s) they would like to toggle. Application is open to login page.

Postcondition: The field(s) that the Writer wanted to toggle have been toggled and saved.

Status: Fields are now Experiences within Experience Categories and Bullets in Bullet Categories and Experiences.

UC12 - Change User Roles

Scope: Resumé Review Software

Level: User goal

Primary Actor: Admin

Stakeholders and Interests:

- Admin: Wishes to change the role of a user, either at their request or for some other reason. Wishes to retain control of who holds what role in the System.
- User: Wishes to be alerted should their role be changed.

Preconditions: The Admin is signed into the system and has found the User they wish to change the role of.

Postcondition: The Admin has changed a specified User's role. The User is aware of the change to their role.

Status: Voided.

UC13 - Send email with PDF

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to have their resume reviewed..
- Reviewer: Wishes to receive the Writer's Resume requesting review.
- Admin: Wishes for the Writer to have their Resume reviewed.

Preconditions: The Writer is signed into the system.

Postcondition: The User's Resume PDF is sent via email to a Reviewer

Status: Unchanged.

UC14 - Preview Resumé

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: wants to view their resumé as a pdf before they export the resumé
- Admin: wants the Writer to preview only their resumé as a pdf

Preconditions: Writer is logged into the System and has entered their resumé information into different categories

Postcondition: The Writer successfully viewed their resumé as a pdf

Status: This process has been merged with Export Resumé (UC06).

UC15 - Import Resume Data File

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to use the Resume Data File they created on another computer.
- Admin: The input file does not contain corrupted data.

Preconditions: The Writer is signed into the system.

Postcondition: The Resume Data File's data is imported and is open in the editor

Status: Unchanged.

UC16 - Export Resume Data File

Scope: Resumé Review Software

Level: User goal

Primary Actor: Writer

Stakeholders and Interests:

- Writer: Wishes to create a Resume Data File for use on another computer.
- Admin: Wishes for the Writer to receive their Resume Data File.

Preconditions: The Writer is signed into the system.

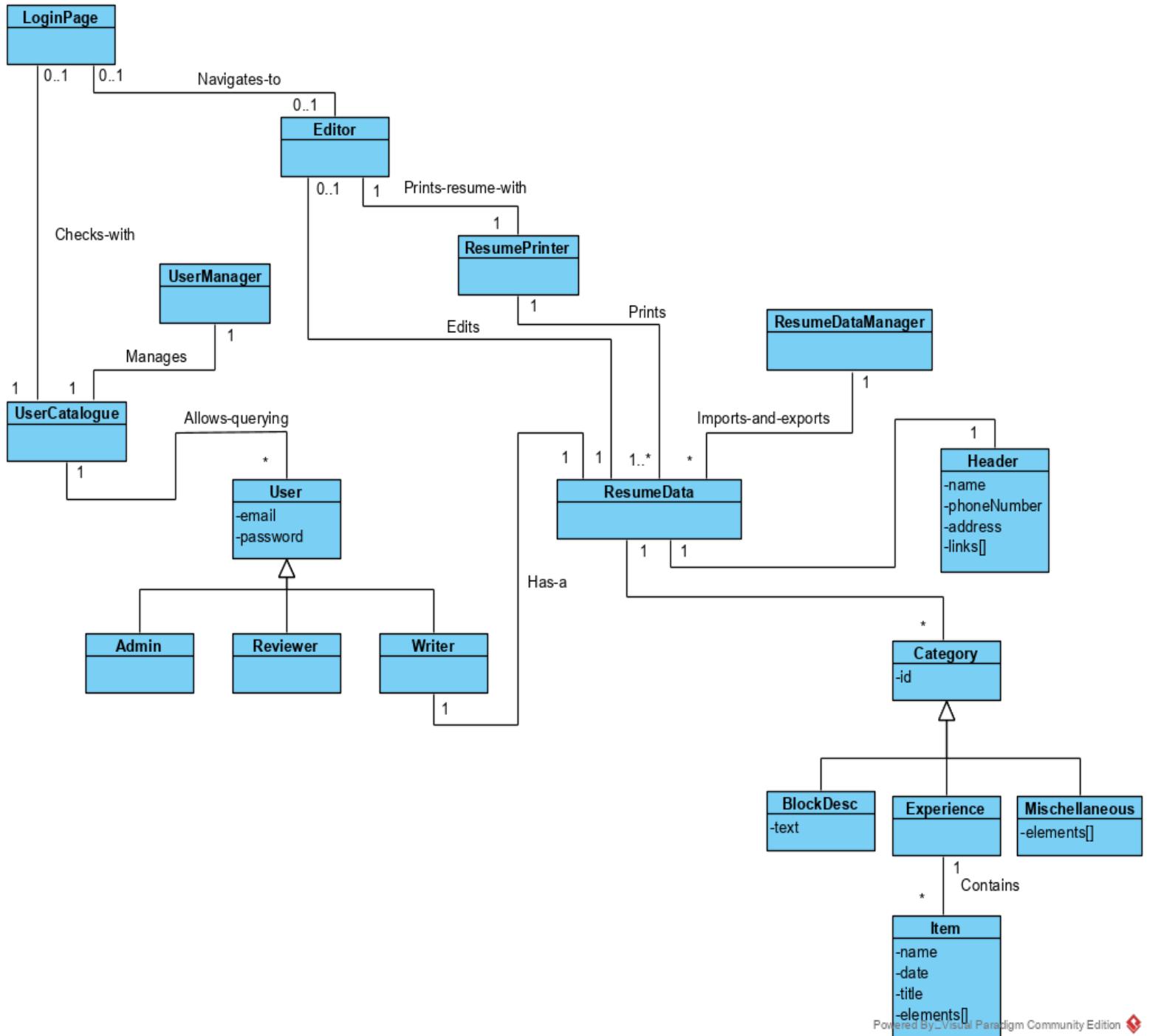
Postcondition: The Resume Data File is exported.

Status: Unchanged.

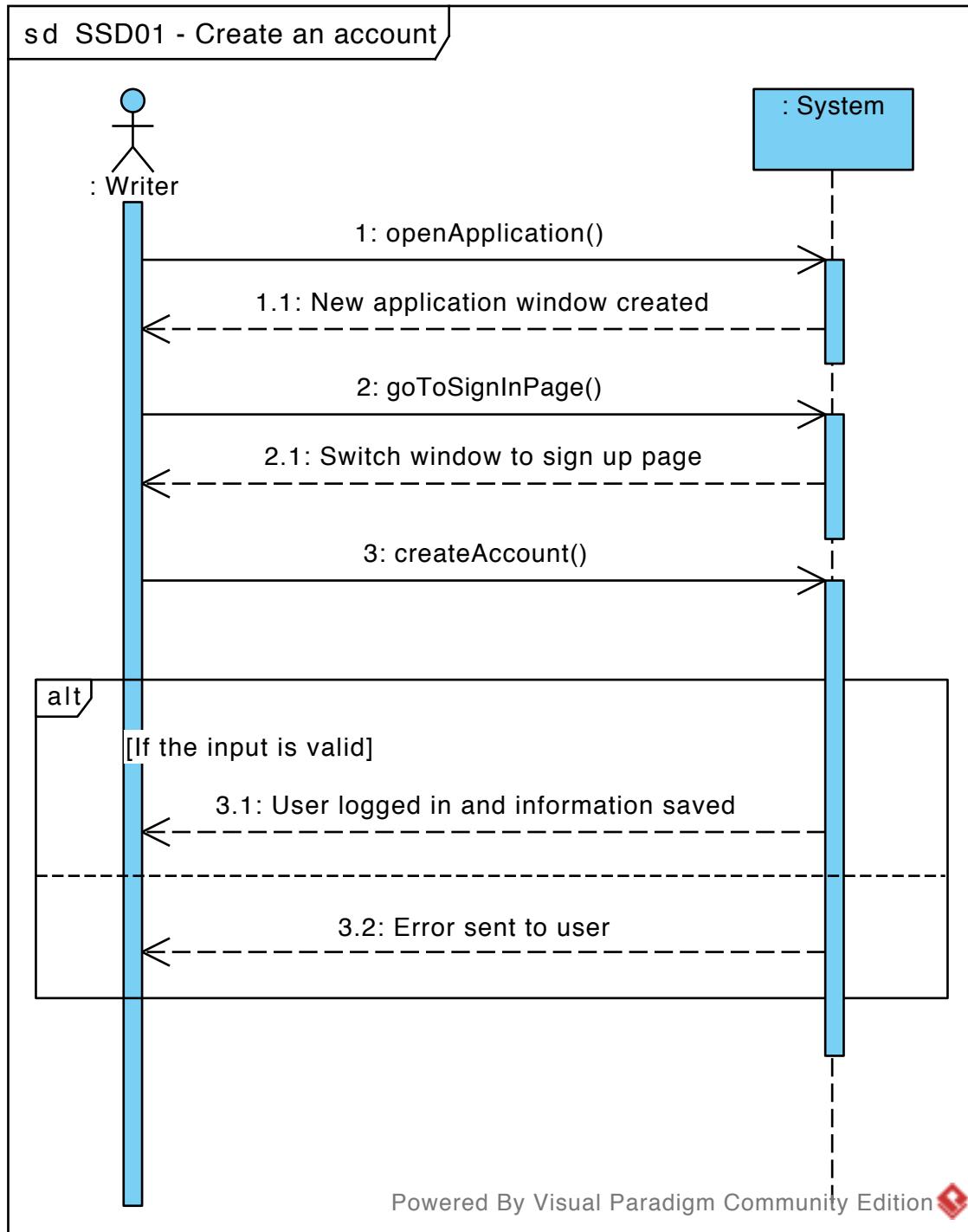
TRACEABILITY MATRIX

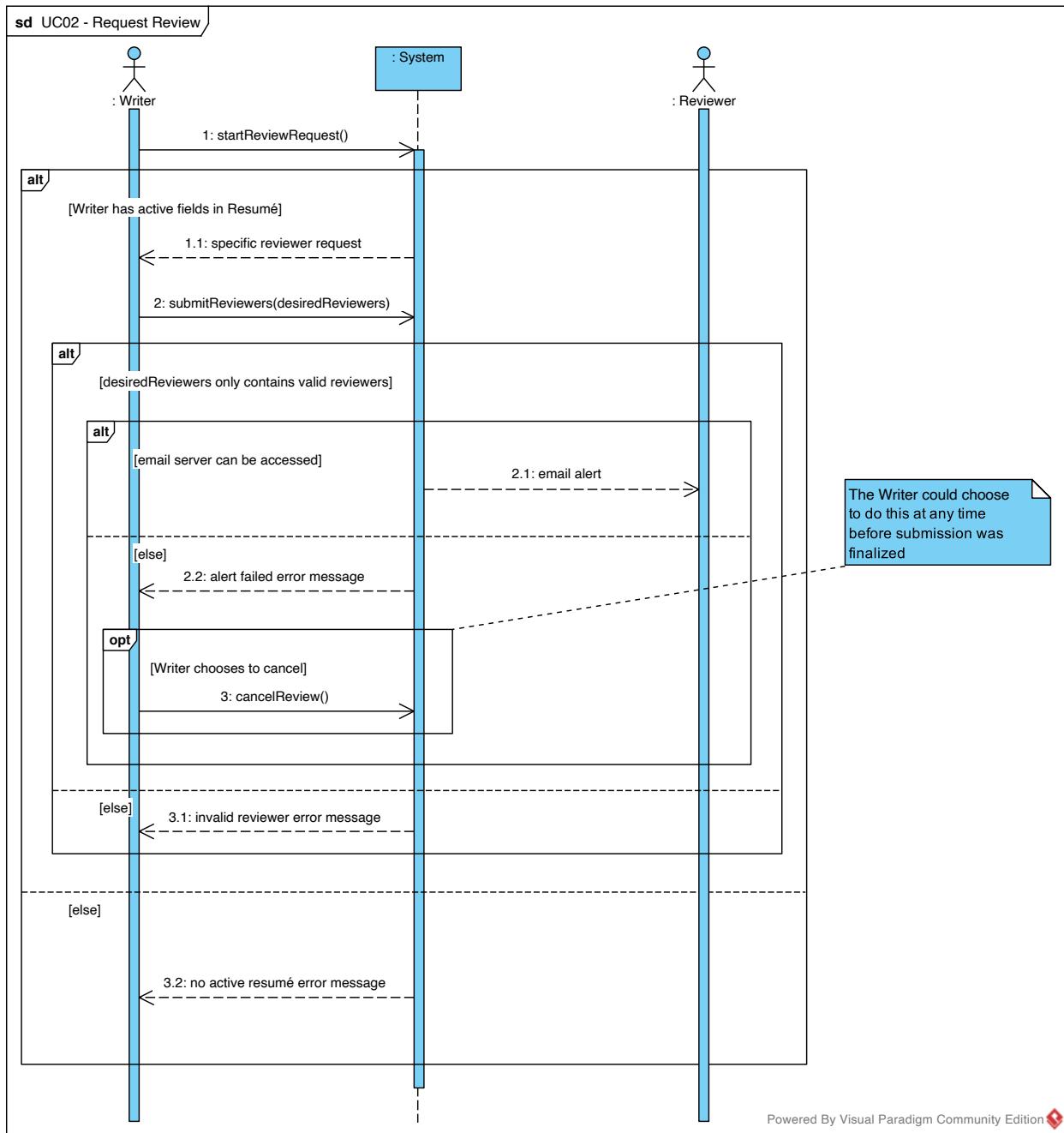
	R01 - Modify resume parts	R02 - Have custom categories	R03 - Modify User Details	R04 - Writer/Review Resume	R05 - Export Resume	R06 - Send review requests	R07 - Have comments	R08 - Toggle resume parts	R09 - User Login
UC01 - Create Account							X		
UC02 - Request Review			X	X			X		
UC03 - Write Review Comments			X		X		X		
UC04 - Edit Items/Fields	X		X				X		
UC05 - Create/Remove Category	X	X	X			X	X		
UC06 - Export Resume				X			X		
UC07 - Edit Header Links/Description	X		X	X			X	X	
UC08 - Check Review Comments			X		X		X	X	
UC09 - Add/Remove Items/Fields	X		X			X	X		
UC10 - Edit Category	X		X				X		
UC11 - Toggle Fields			X			X	X		
UC12 - Change User Role		X	X				X		
UC13 - Send Email With PDF			X	X	X		X		
UC14 - Preview Resume				X			X		
UC15 - Import Resume Data File					X		X		
UC16 - Export Desume Data File					X	X		X	
UC17 - Toggle Category	X					X	X		
UC18 - Modify User Details		X	X						

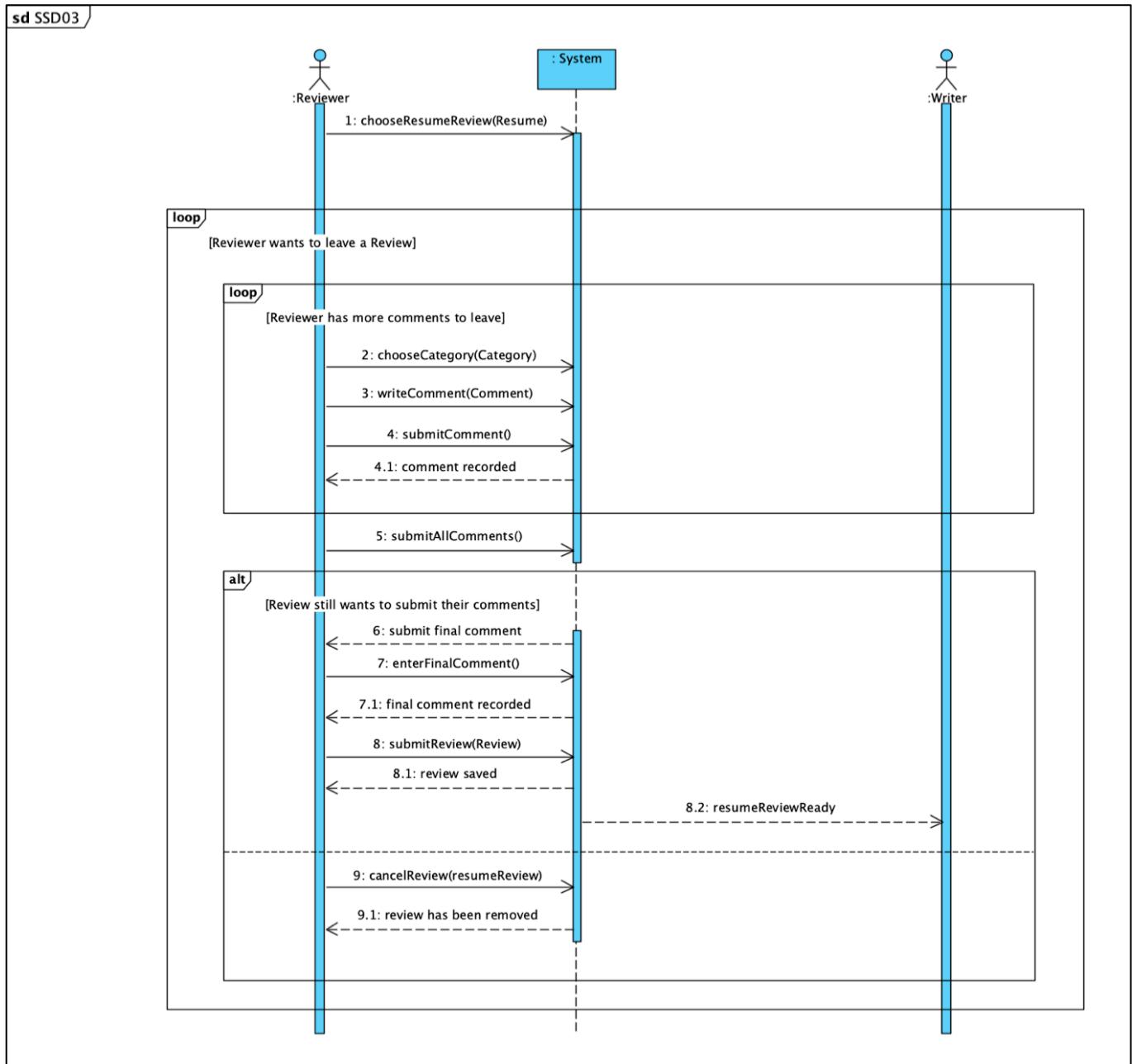
DOMAIN MODEL

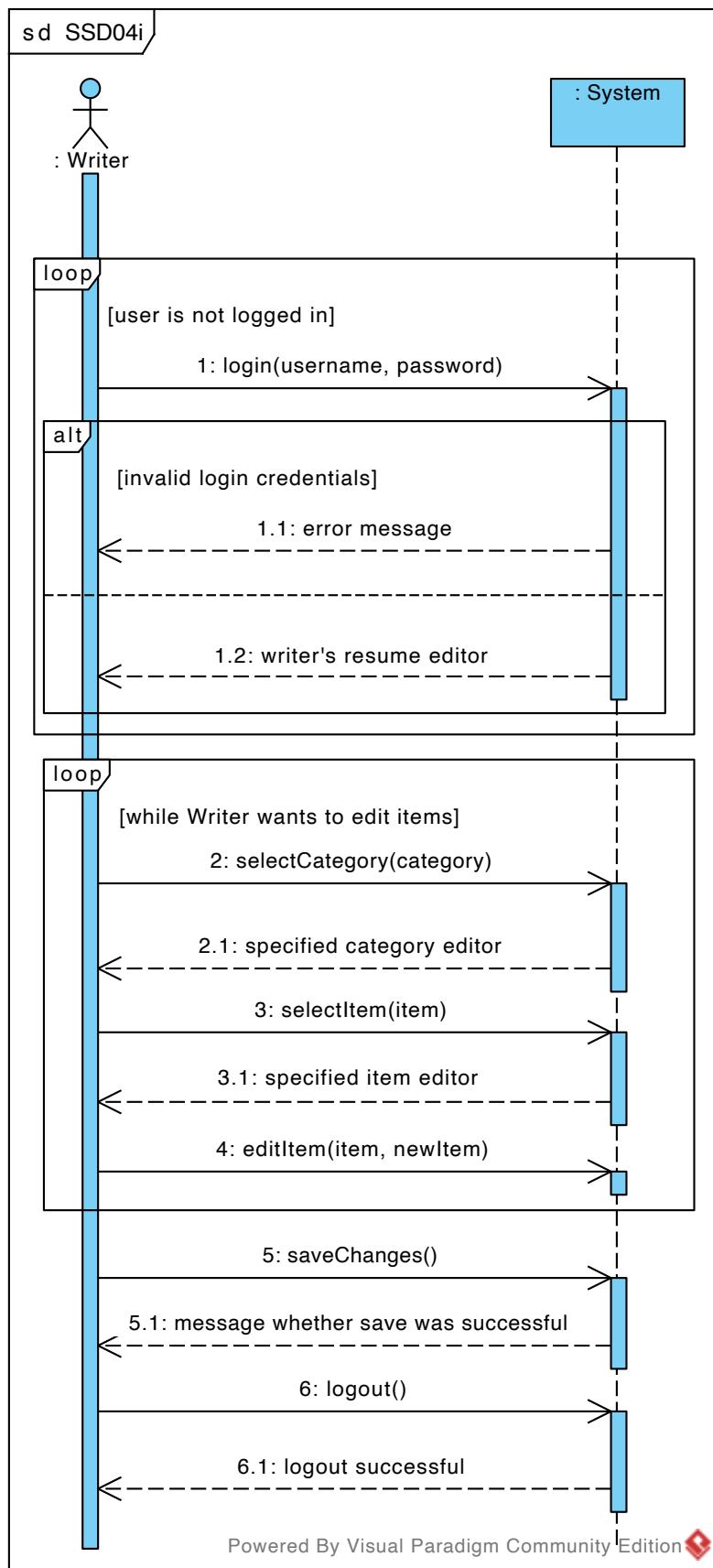


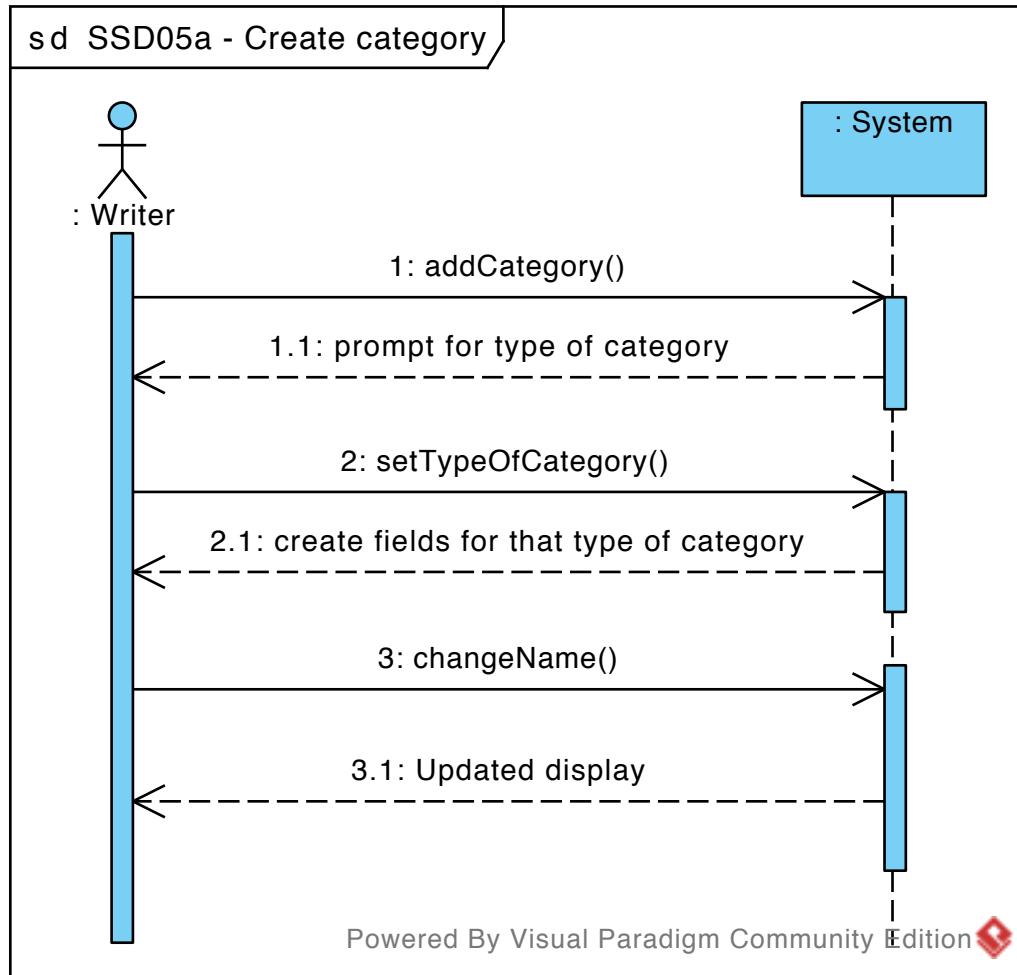
SYSTEM SEQUENCE DIAGRAMS

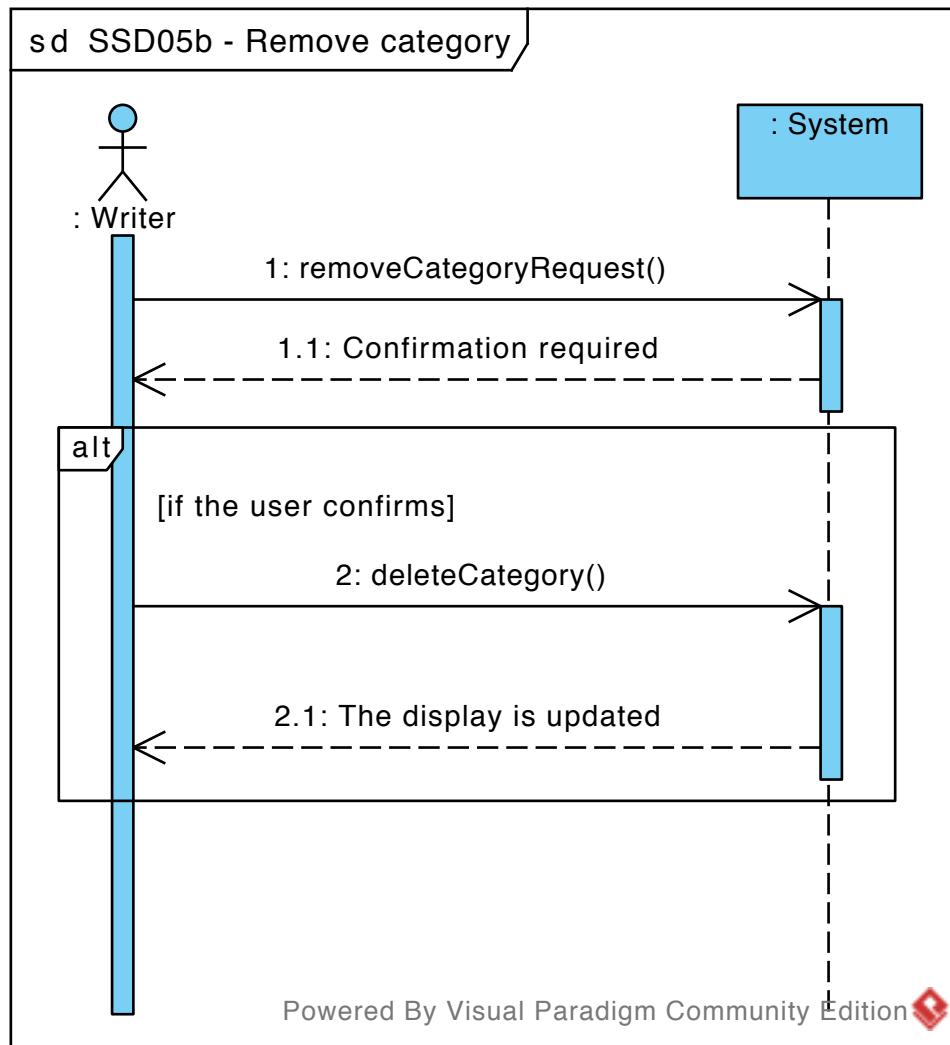


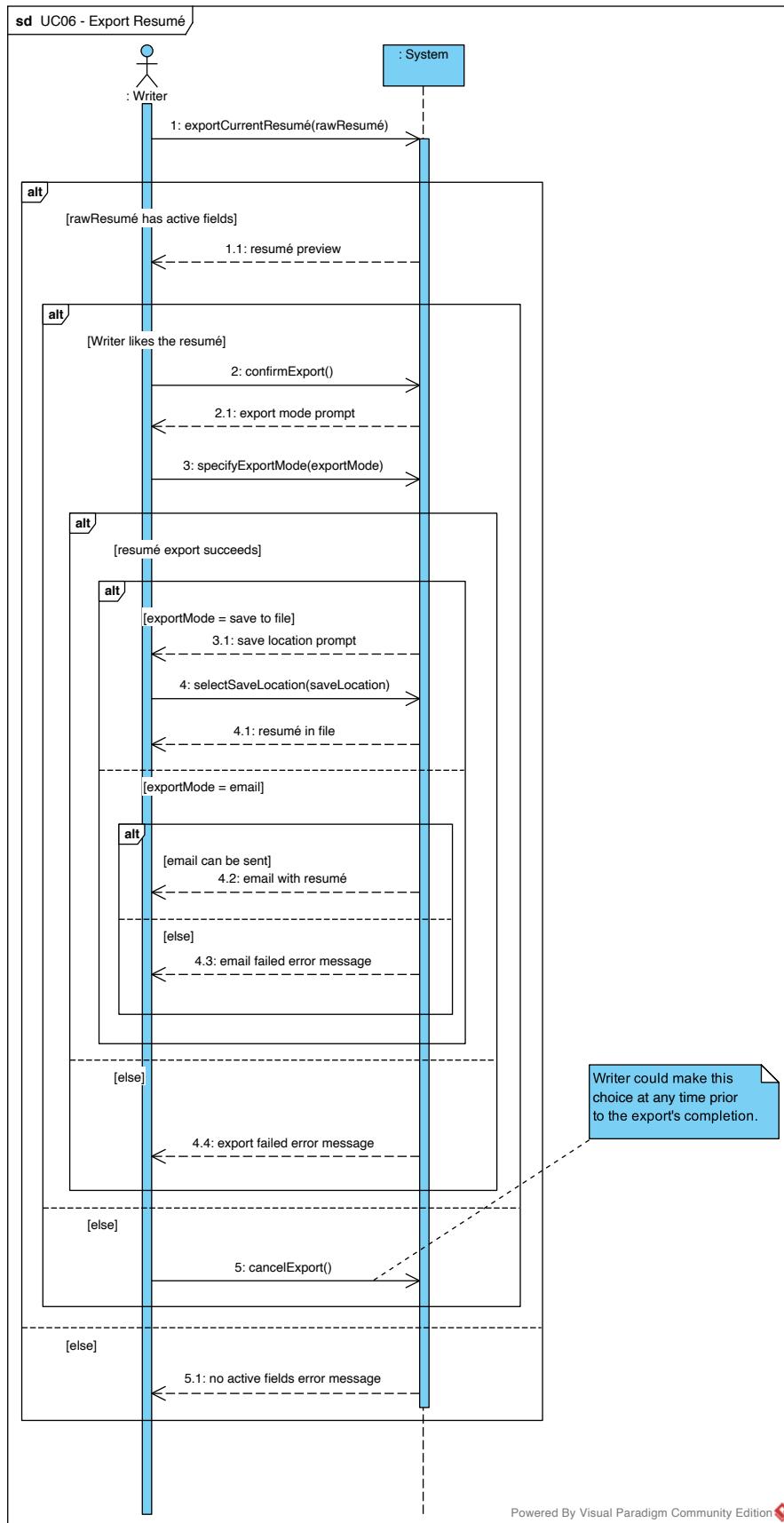


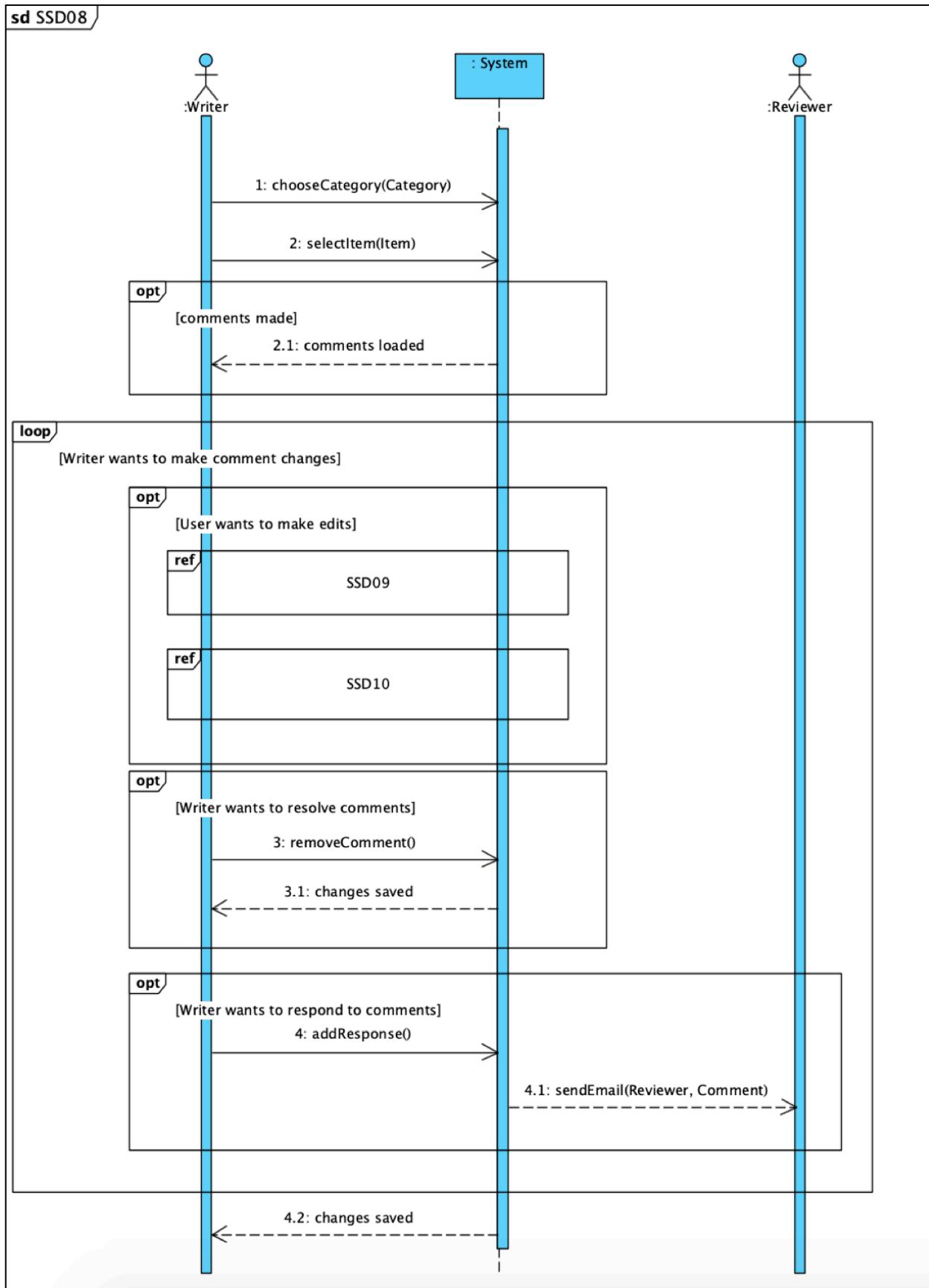


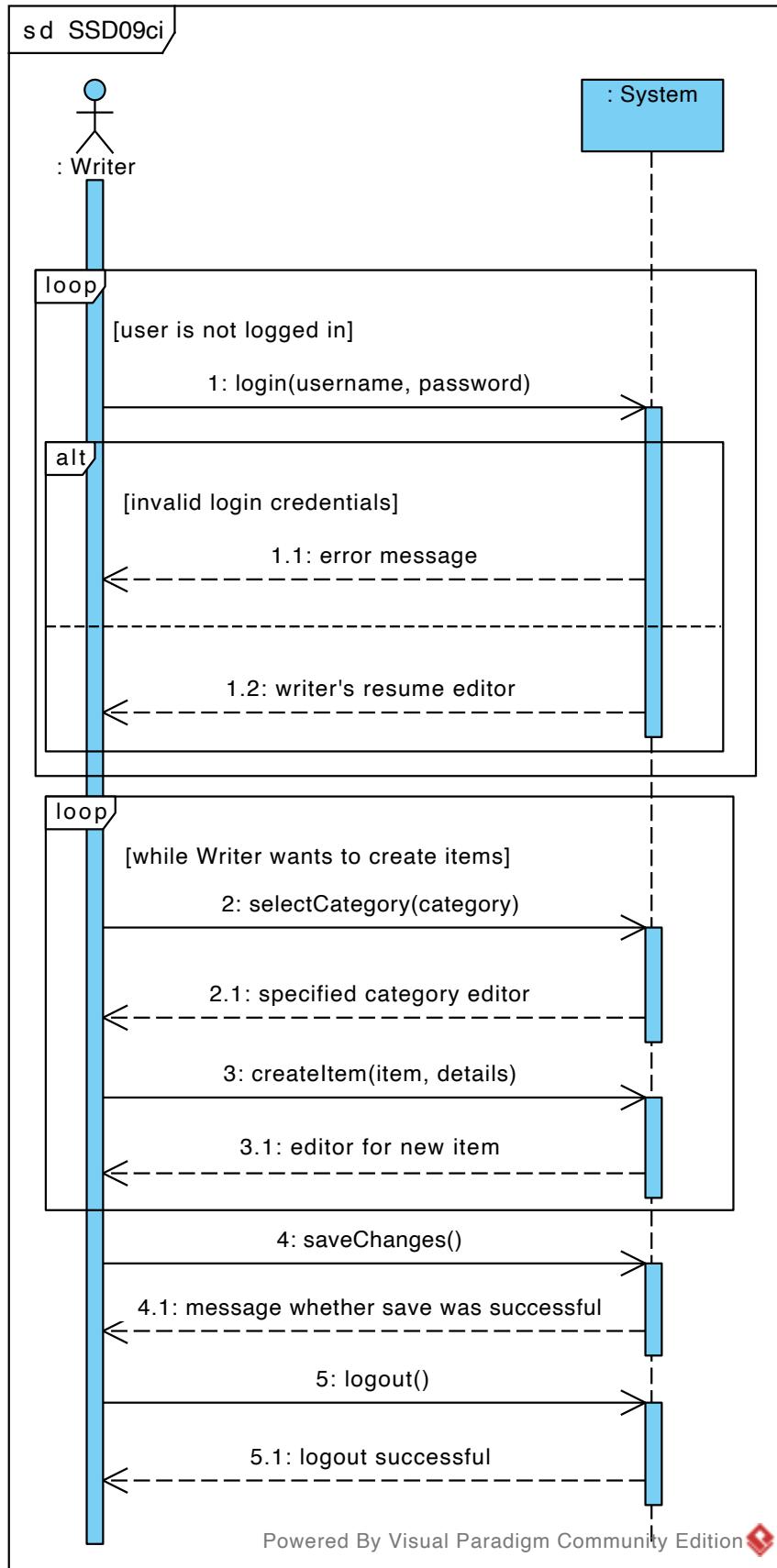


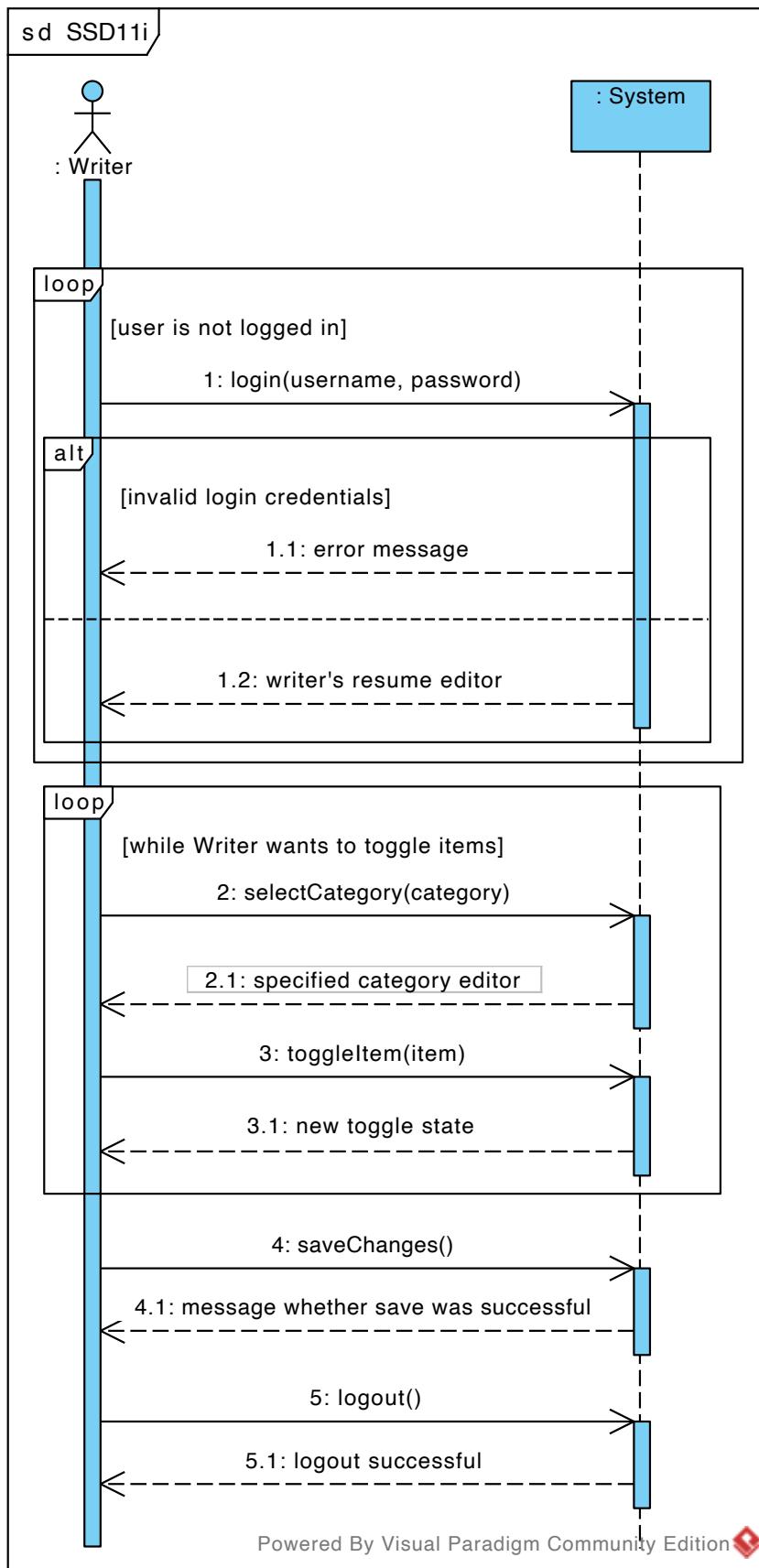


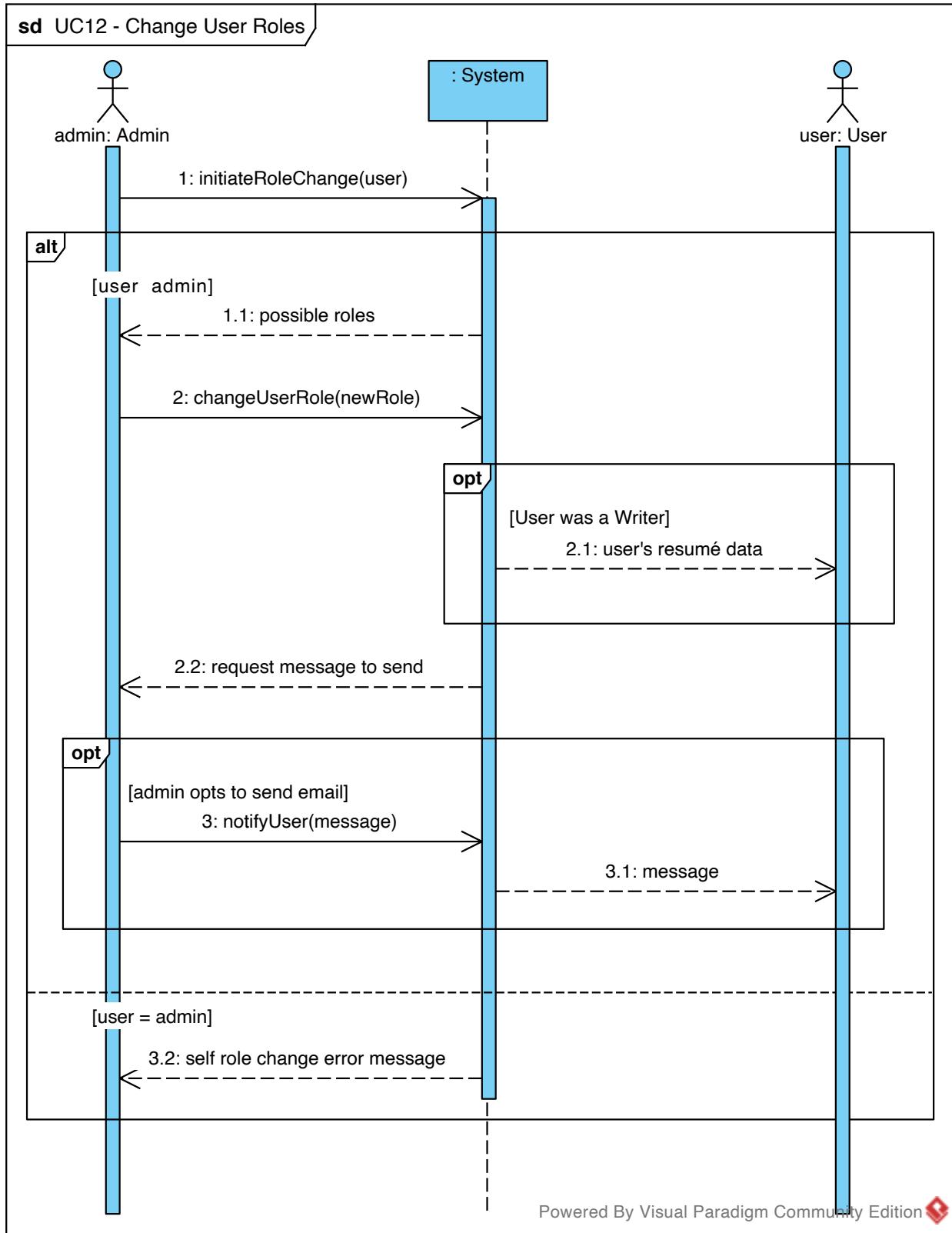


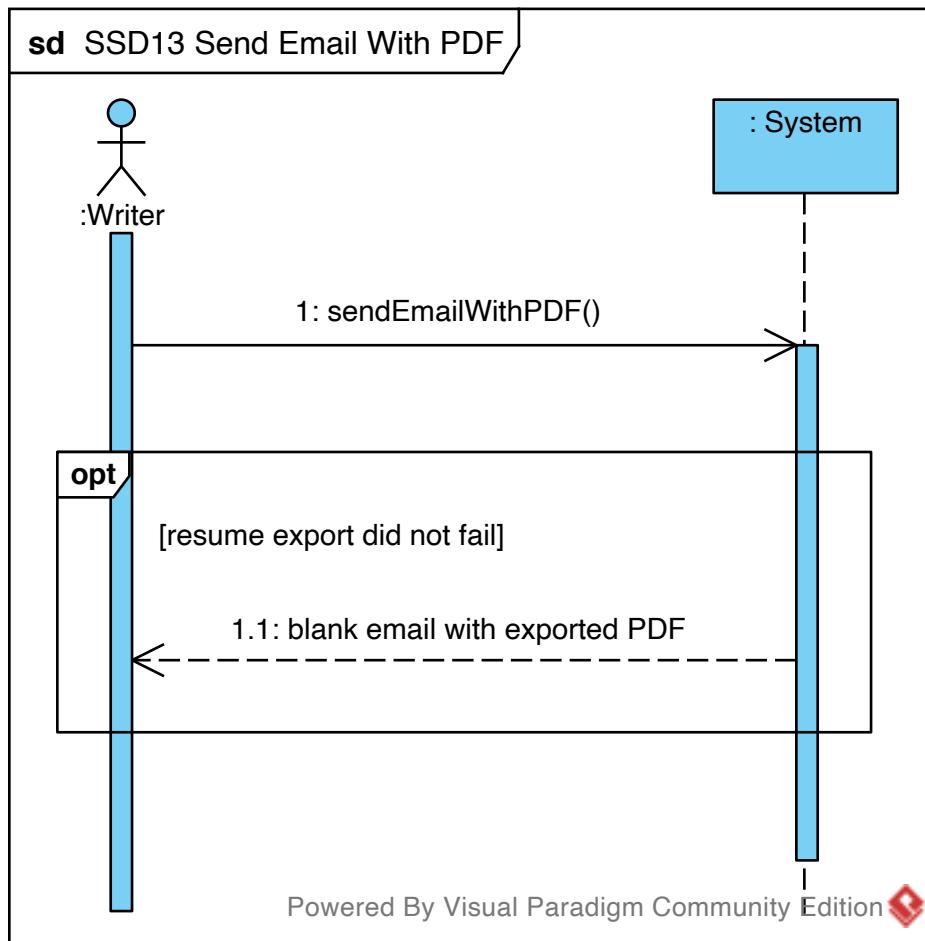


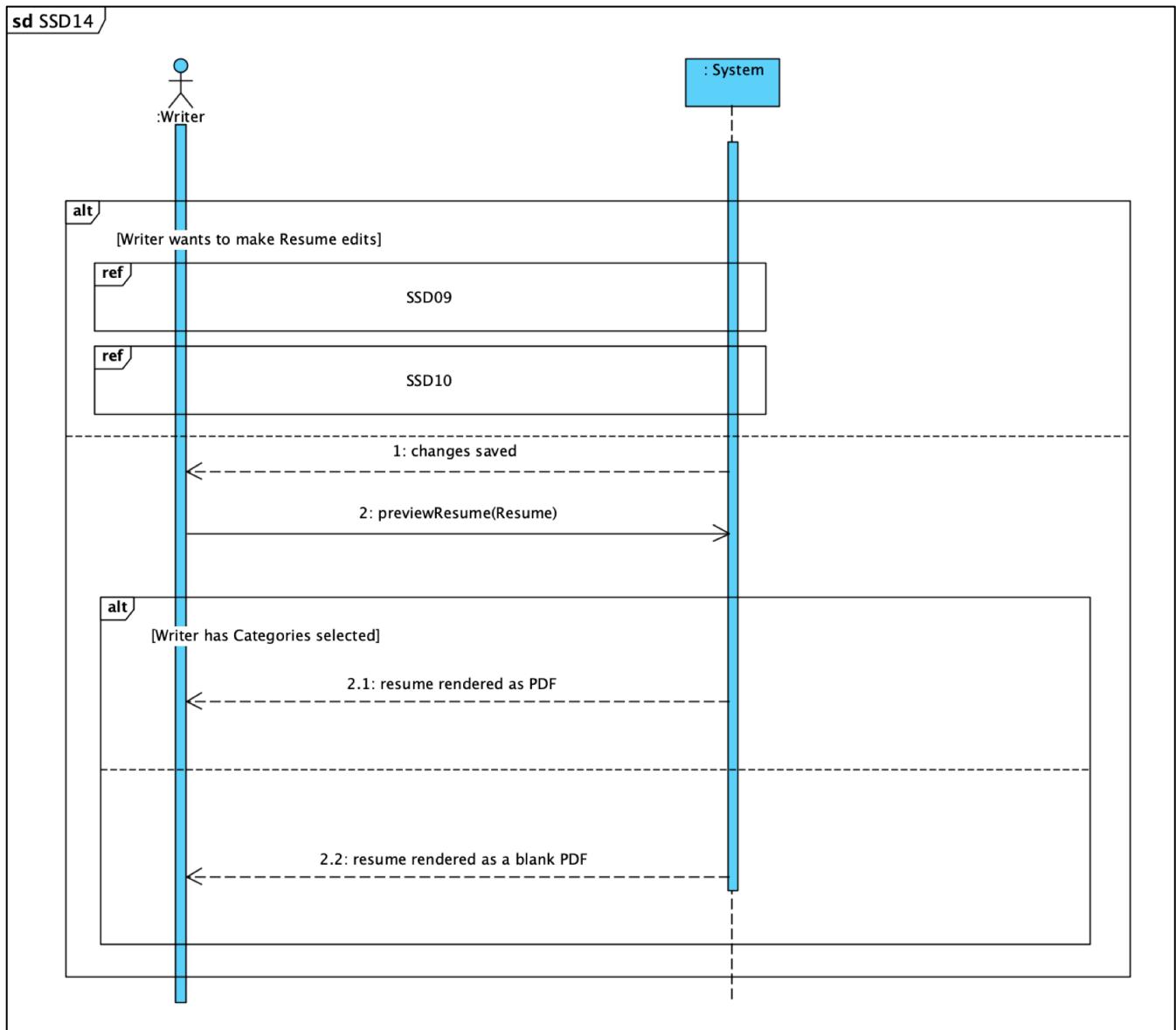


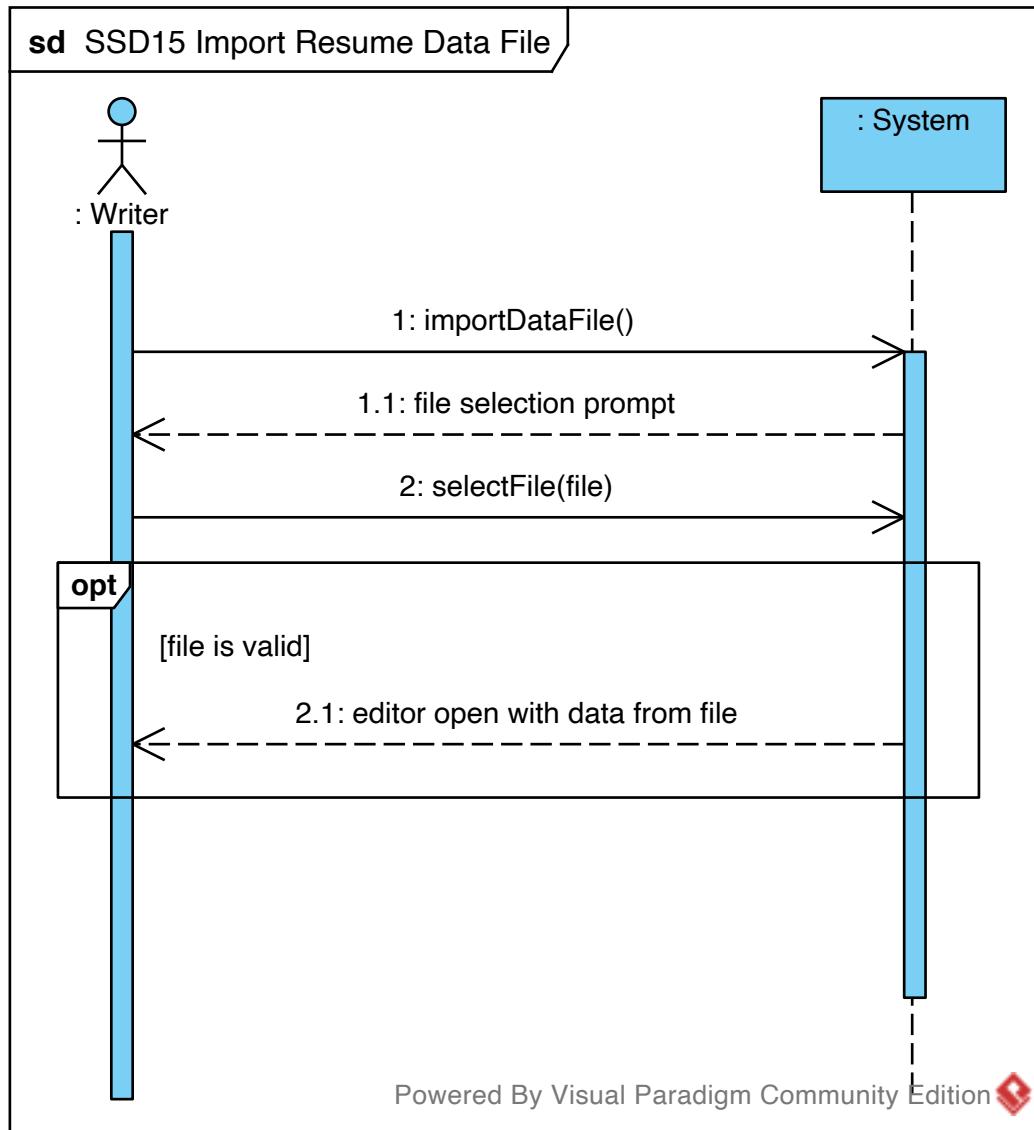


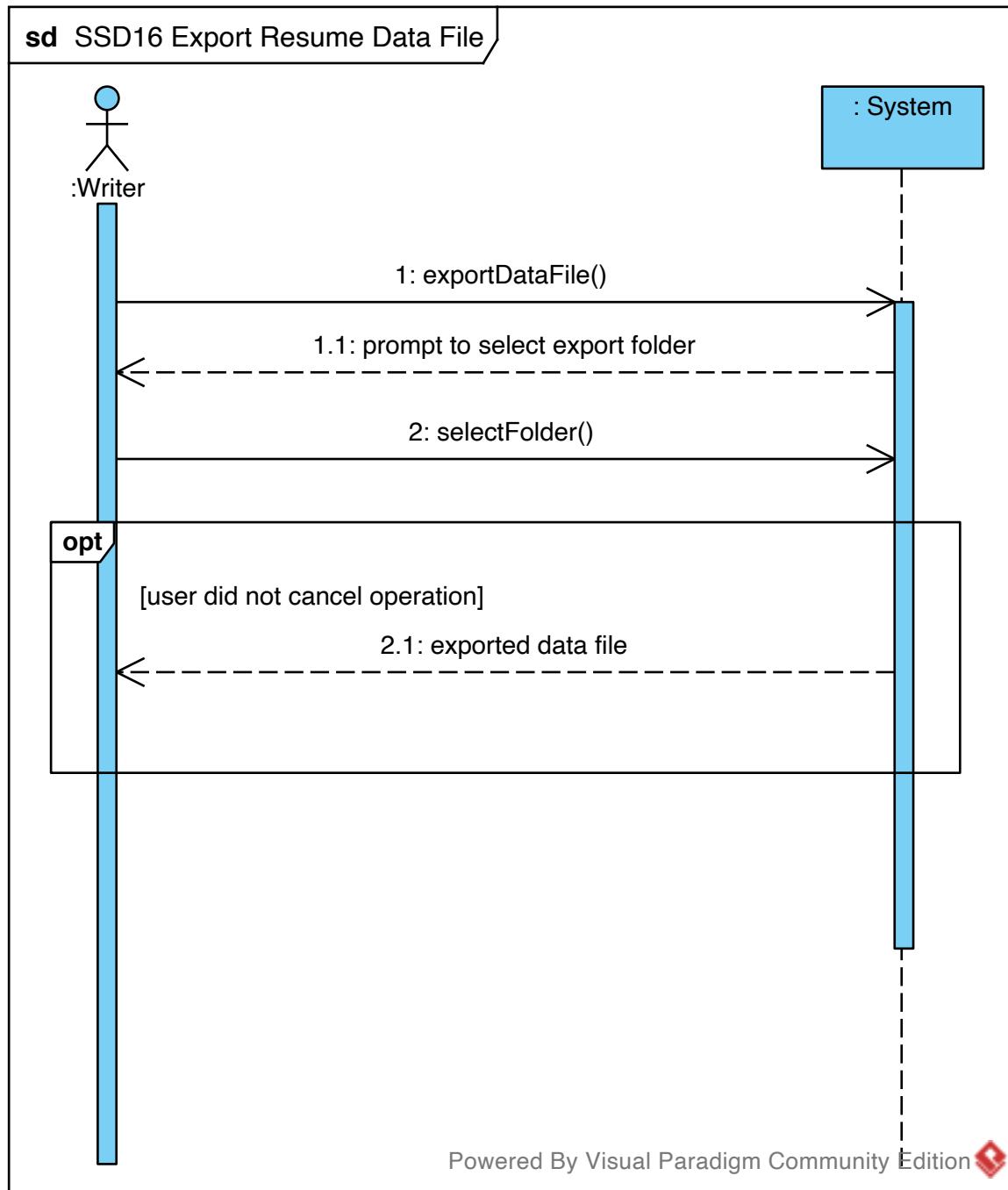






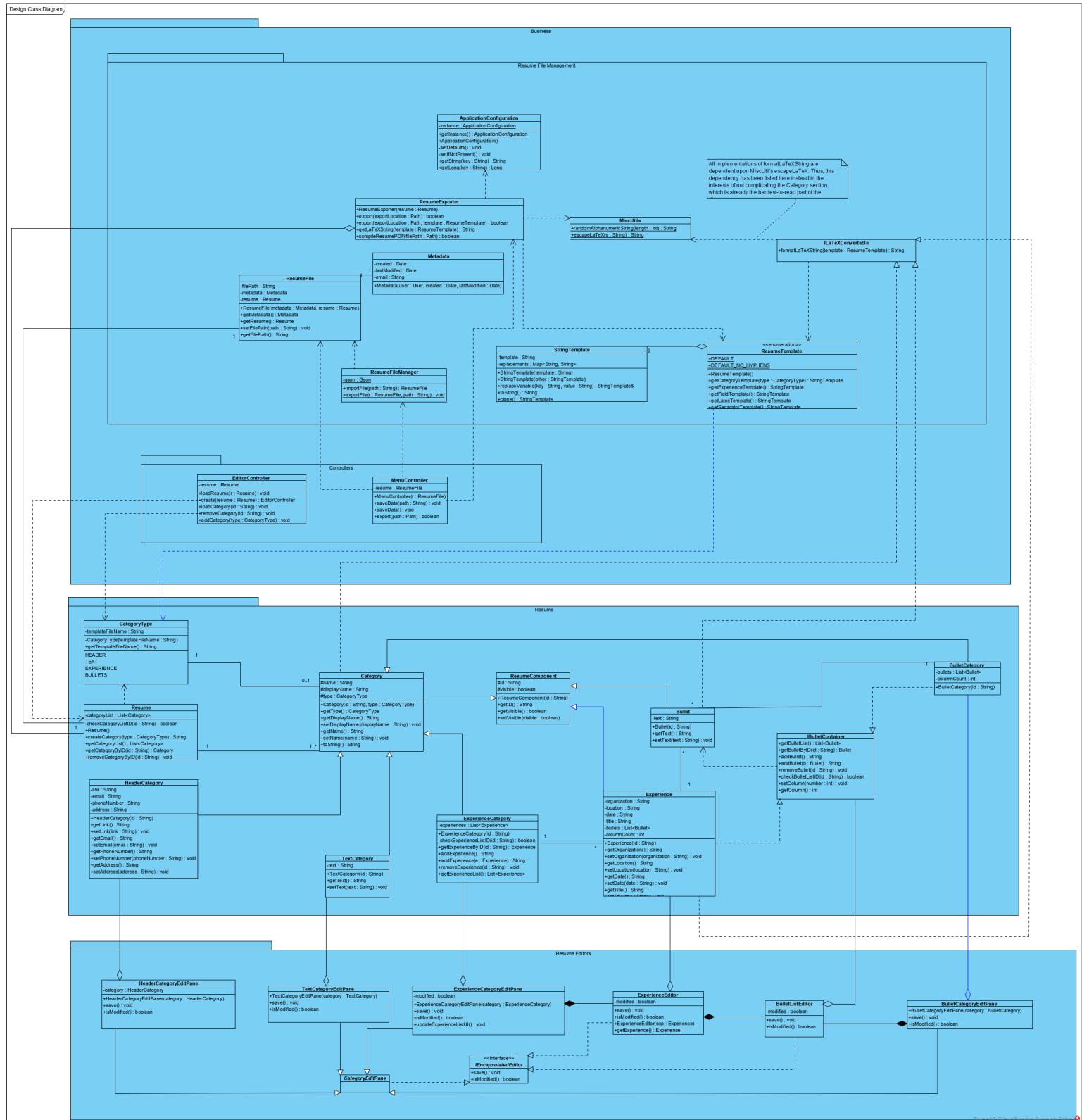




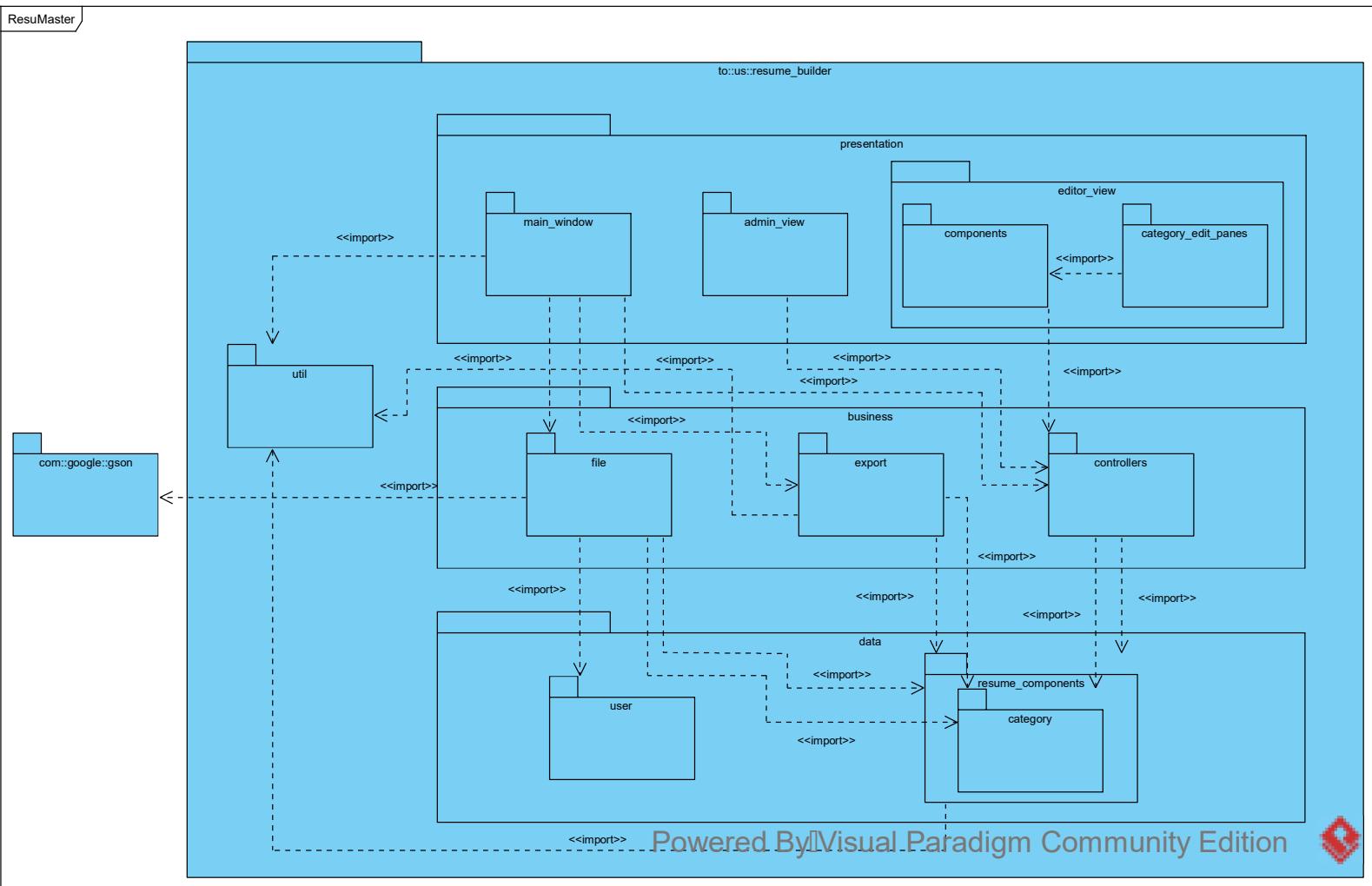


DESIGN

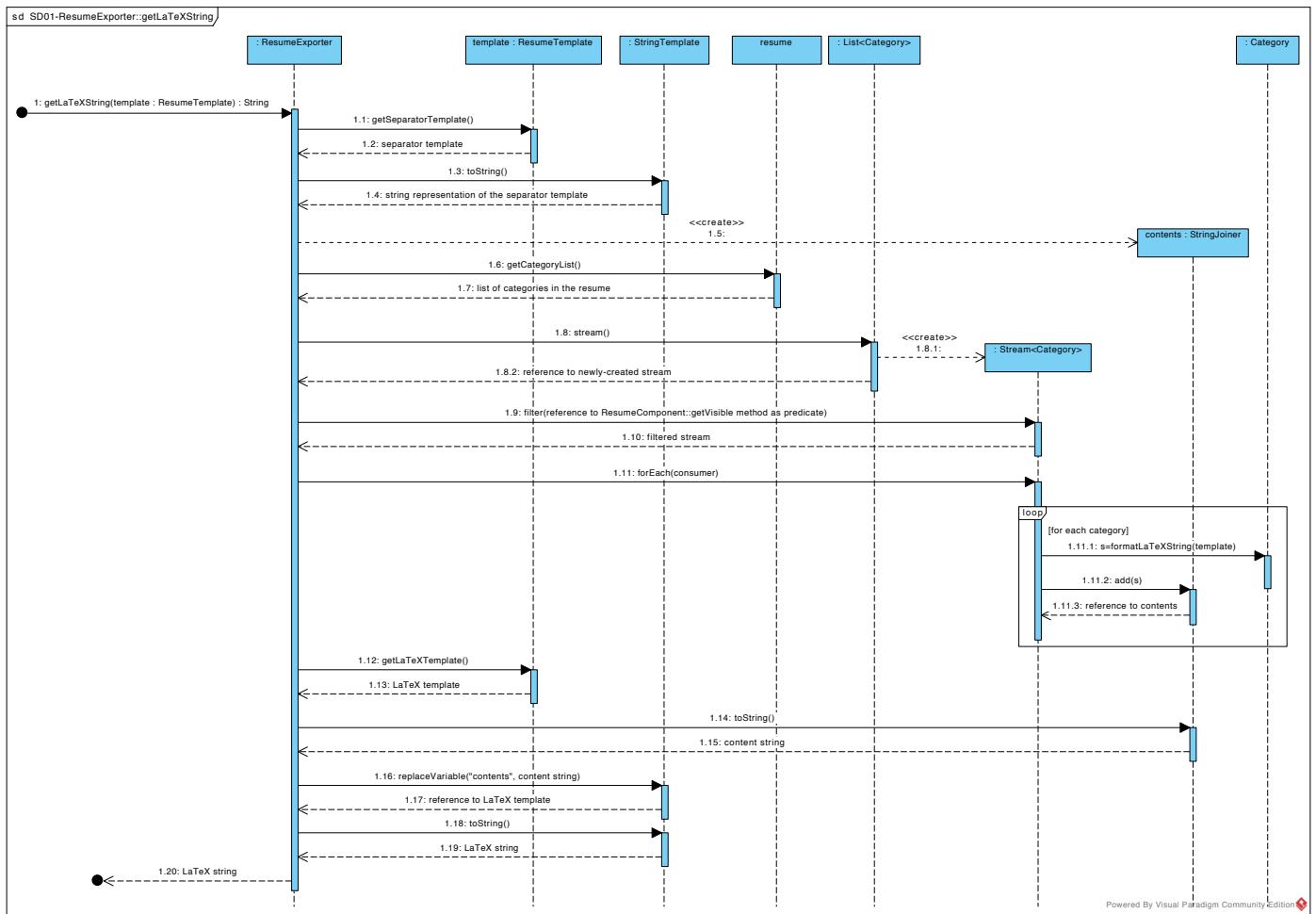
DESIGN CLASS DIAGRAM

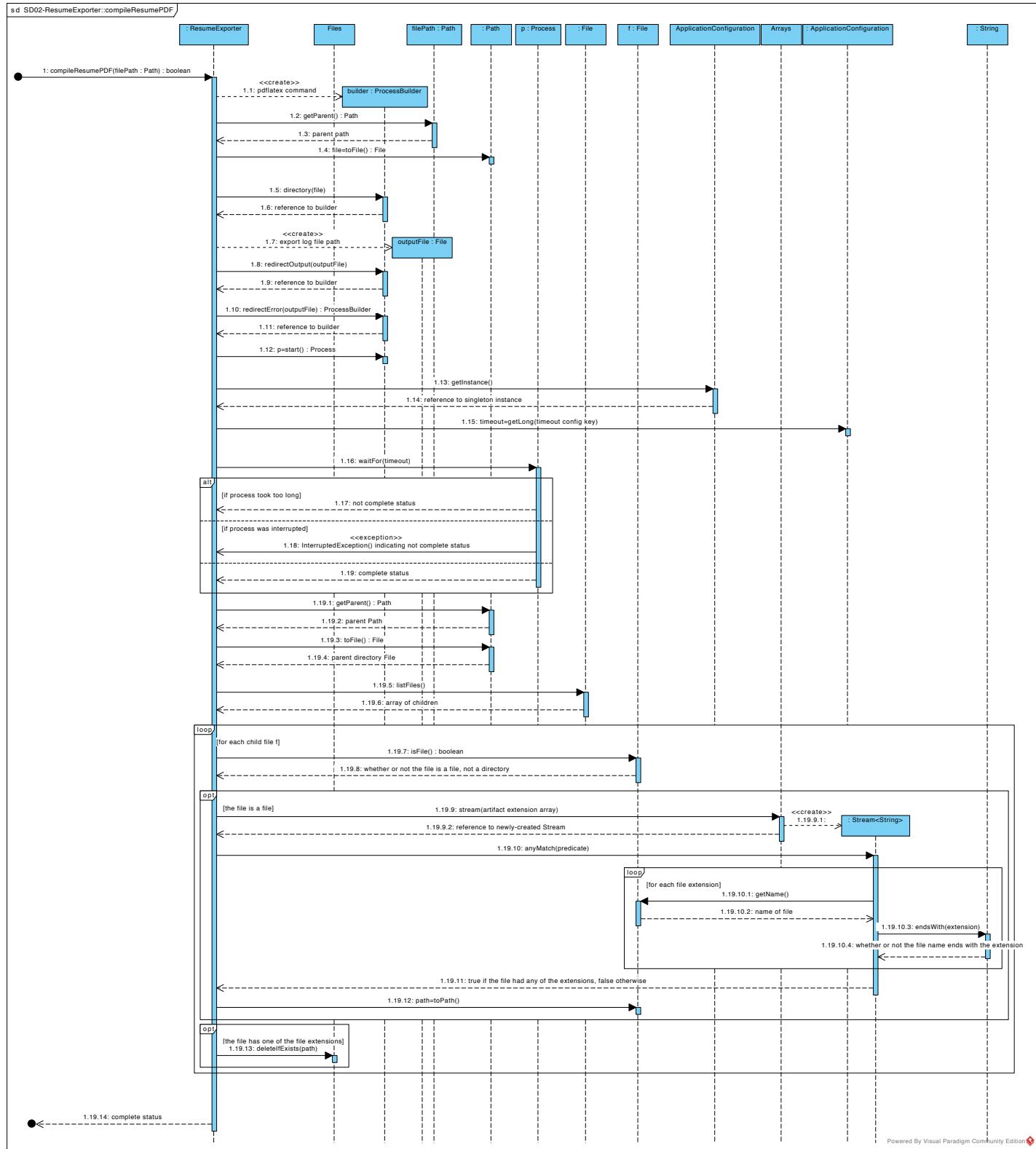


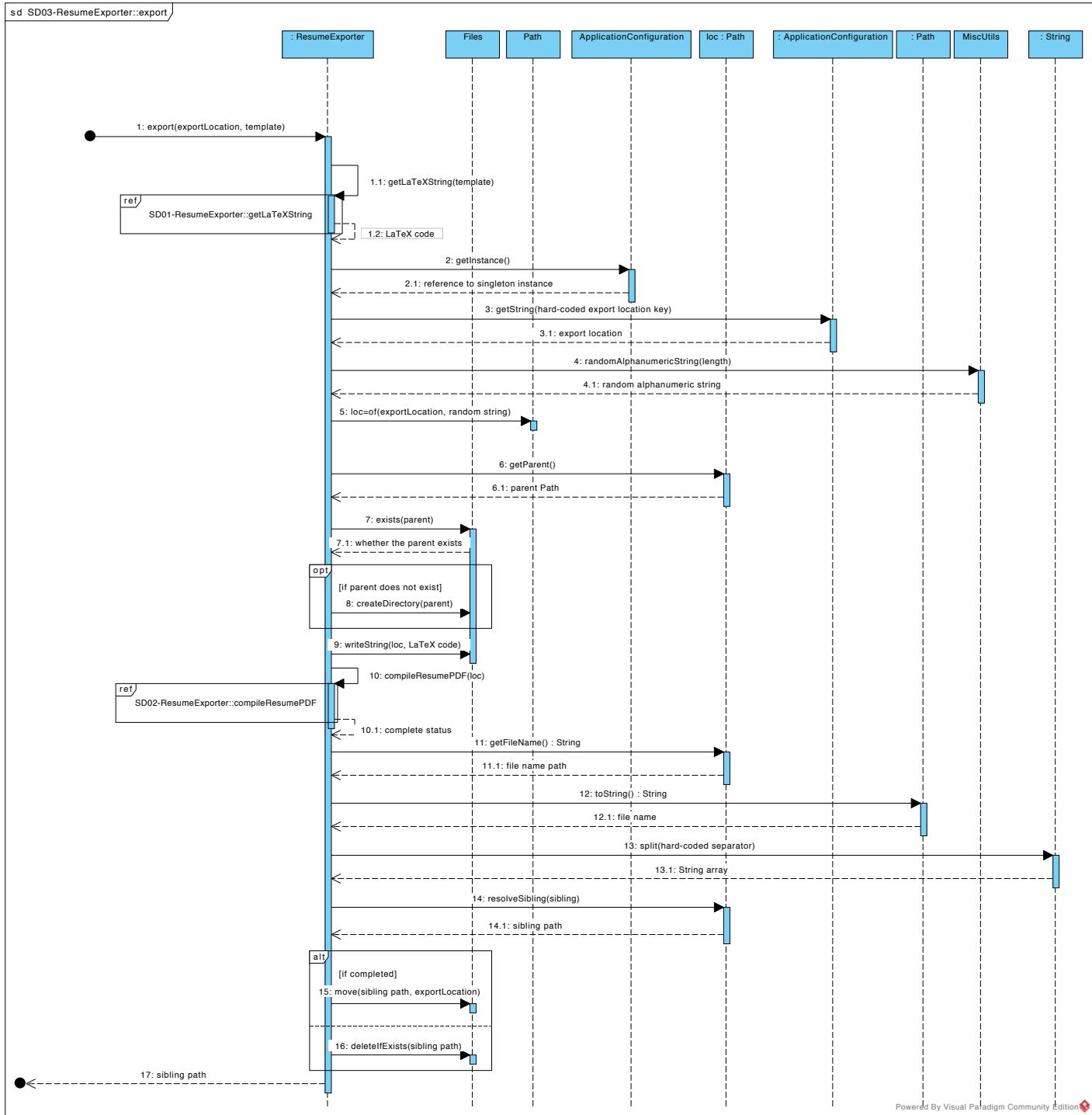
PACKAGE DIAGRAM

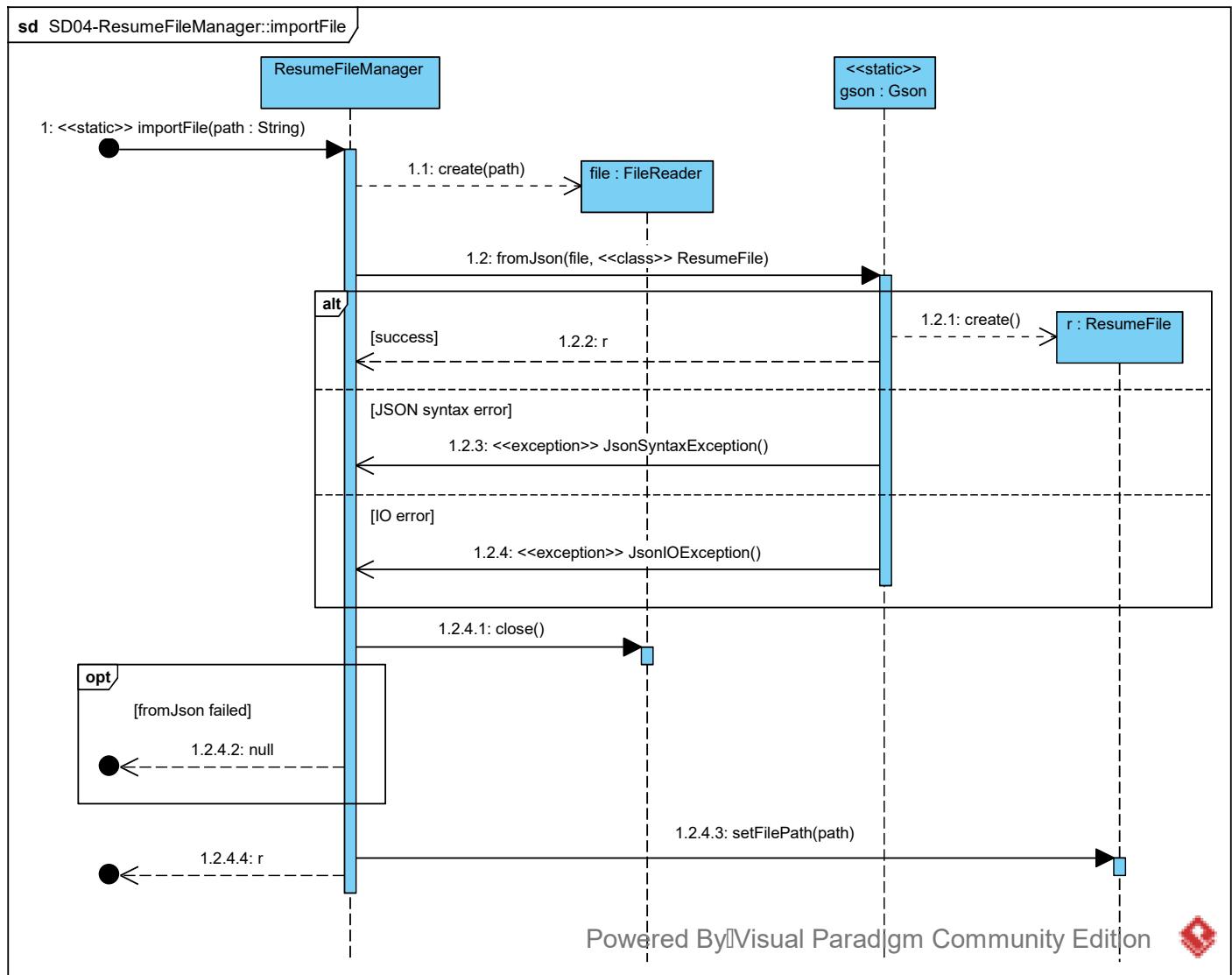


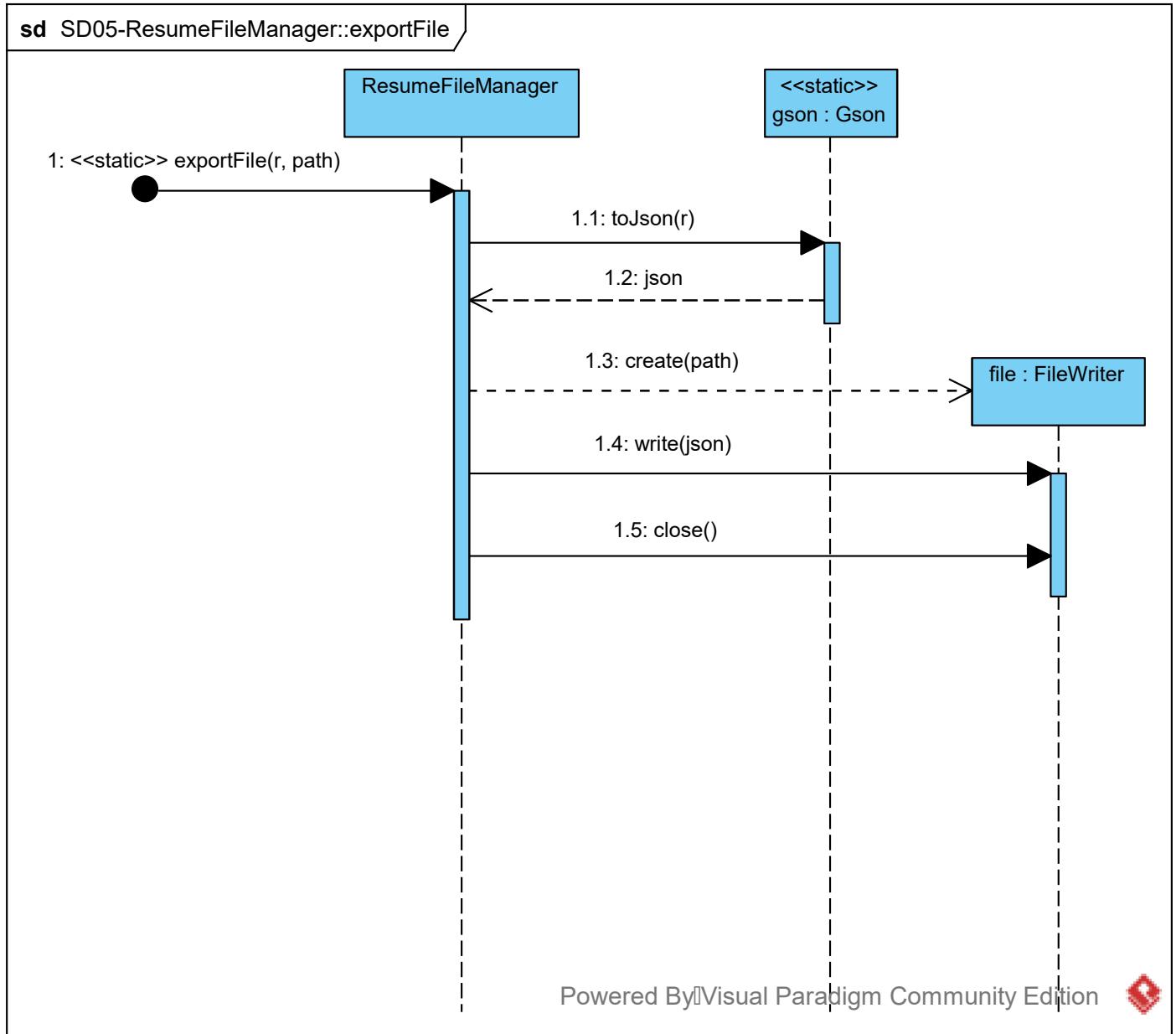
SEQUENCE DIAGRAMS

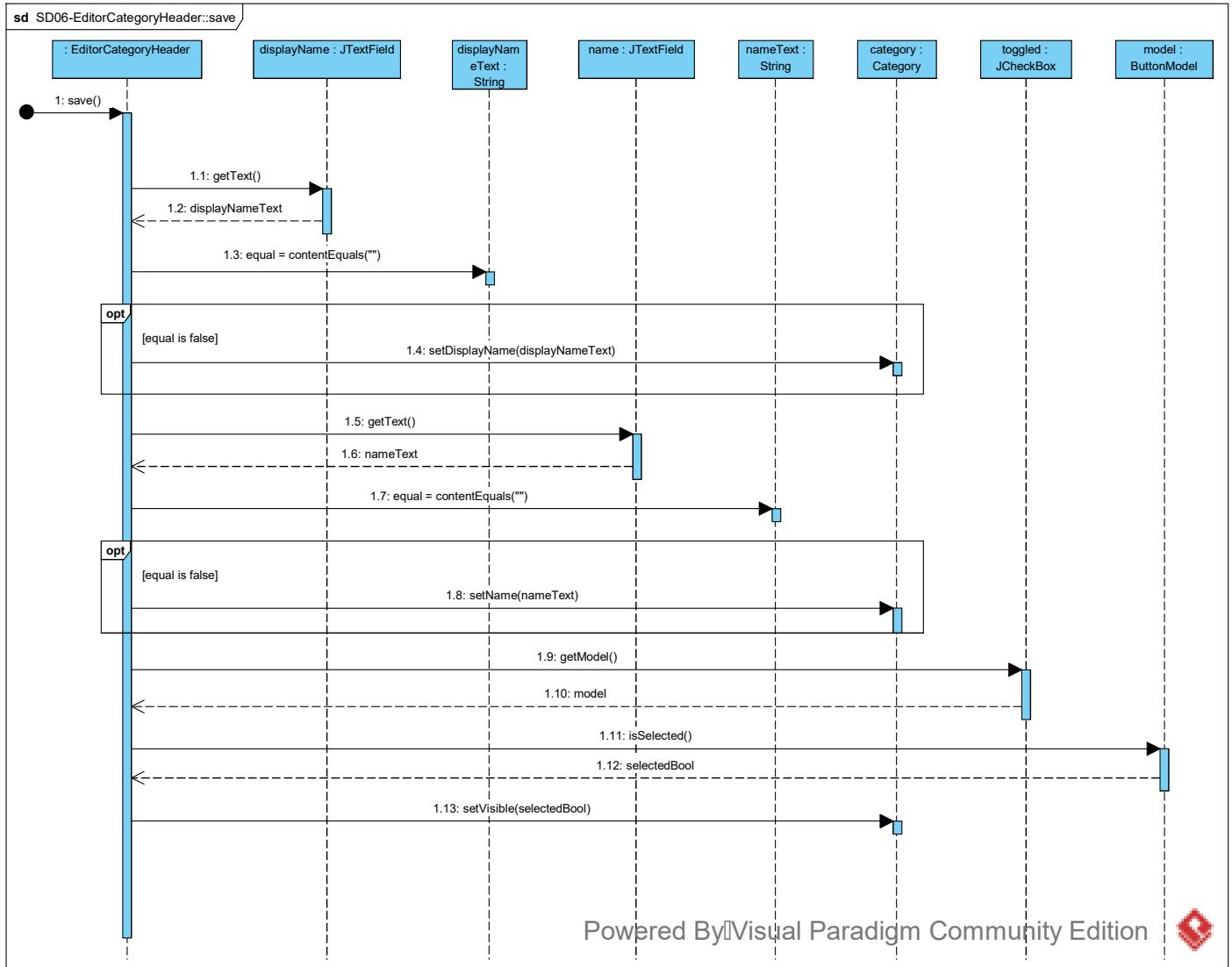


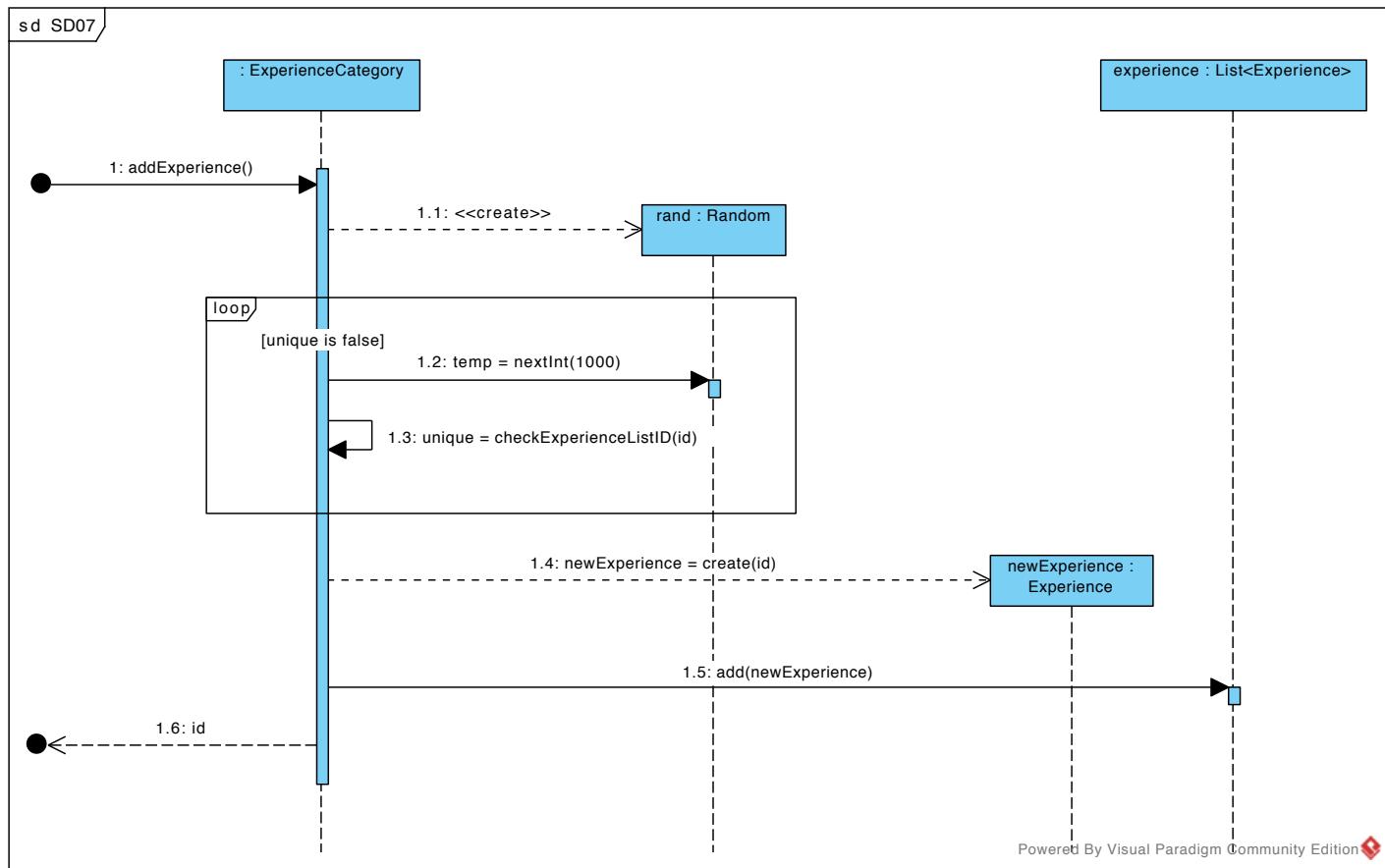


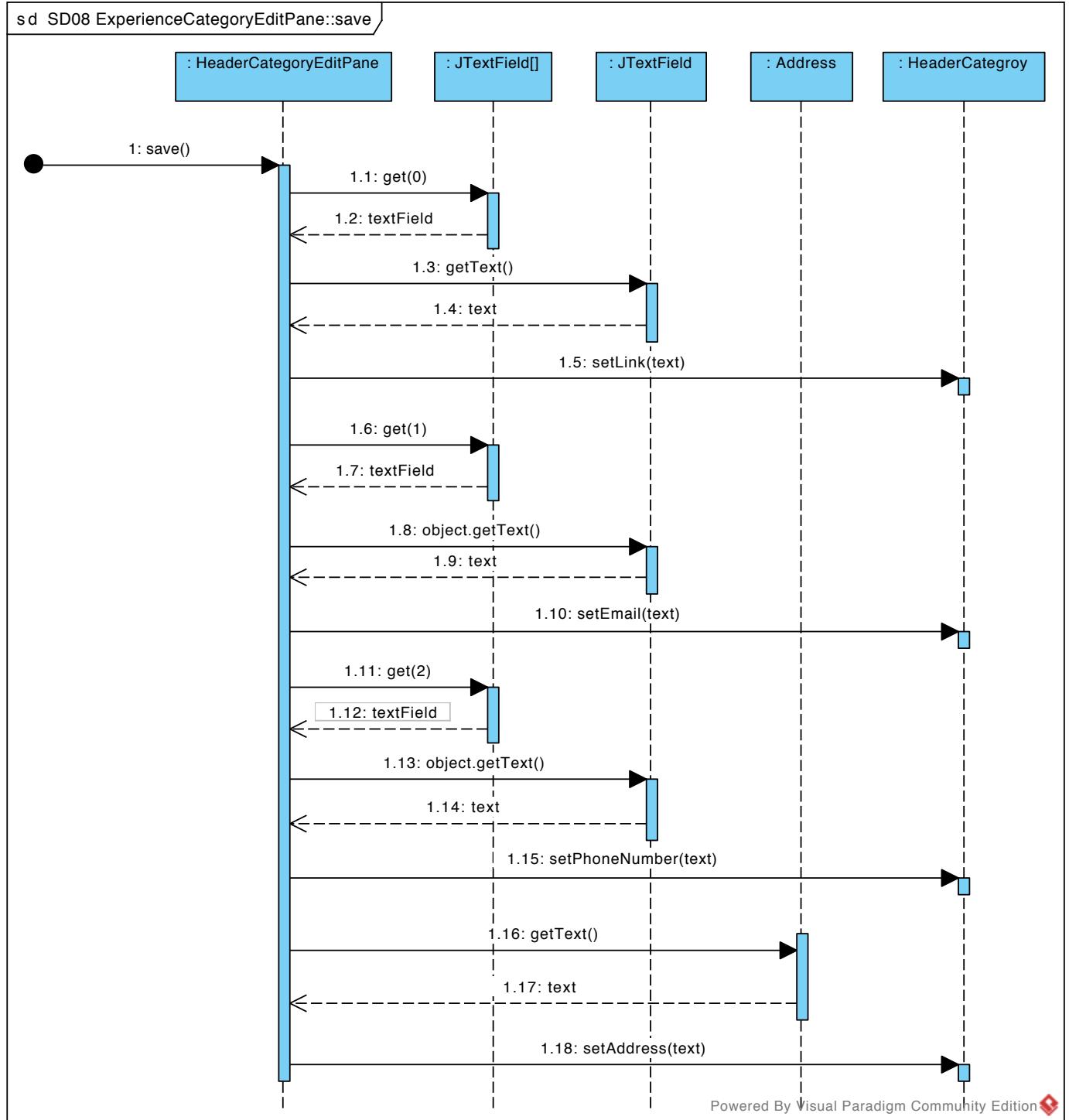


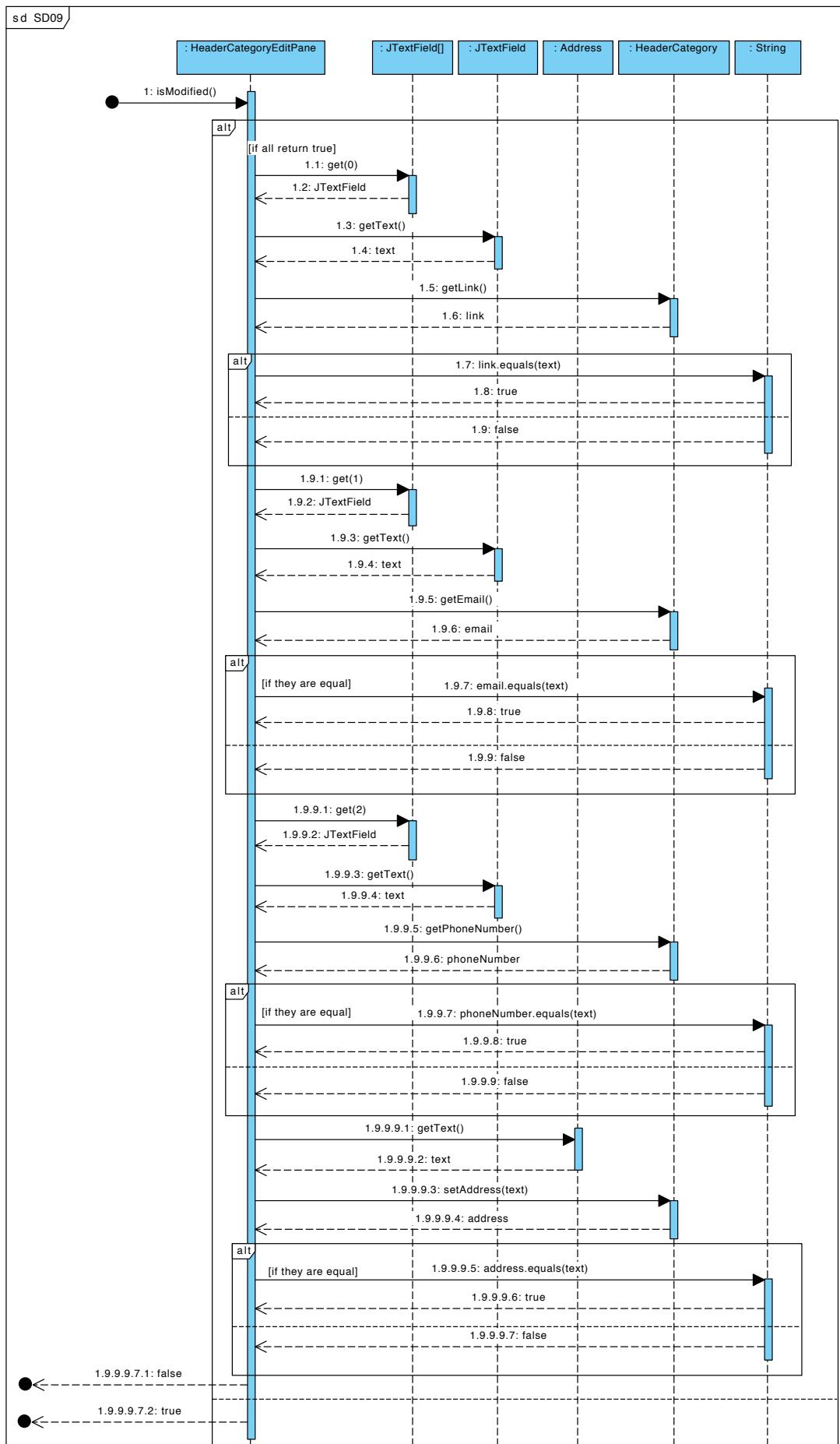


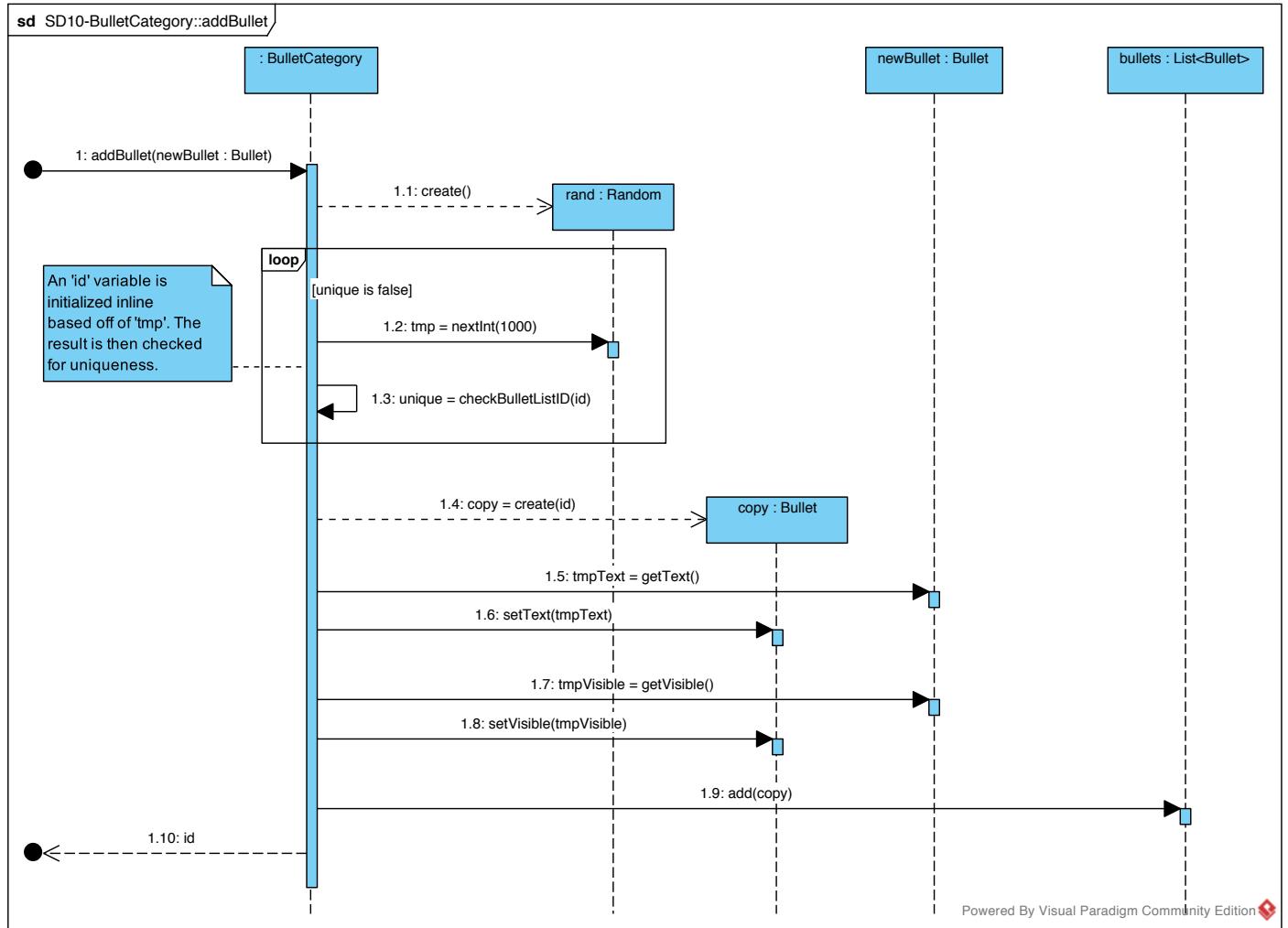


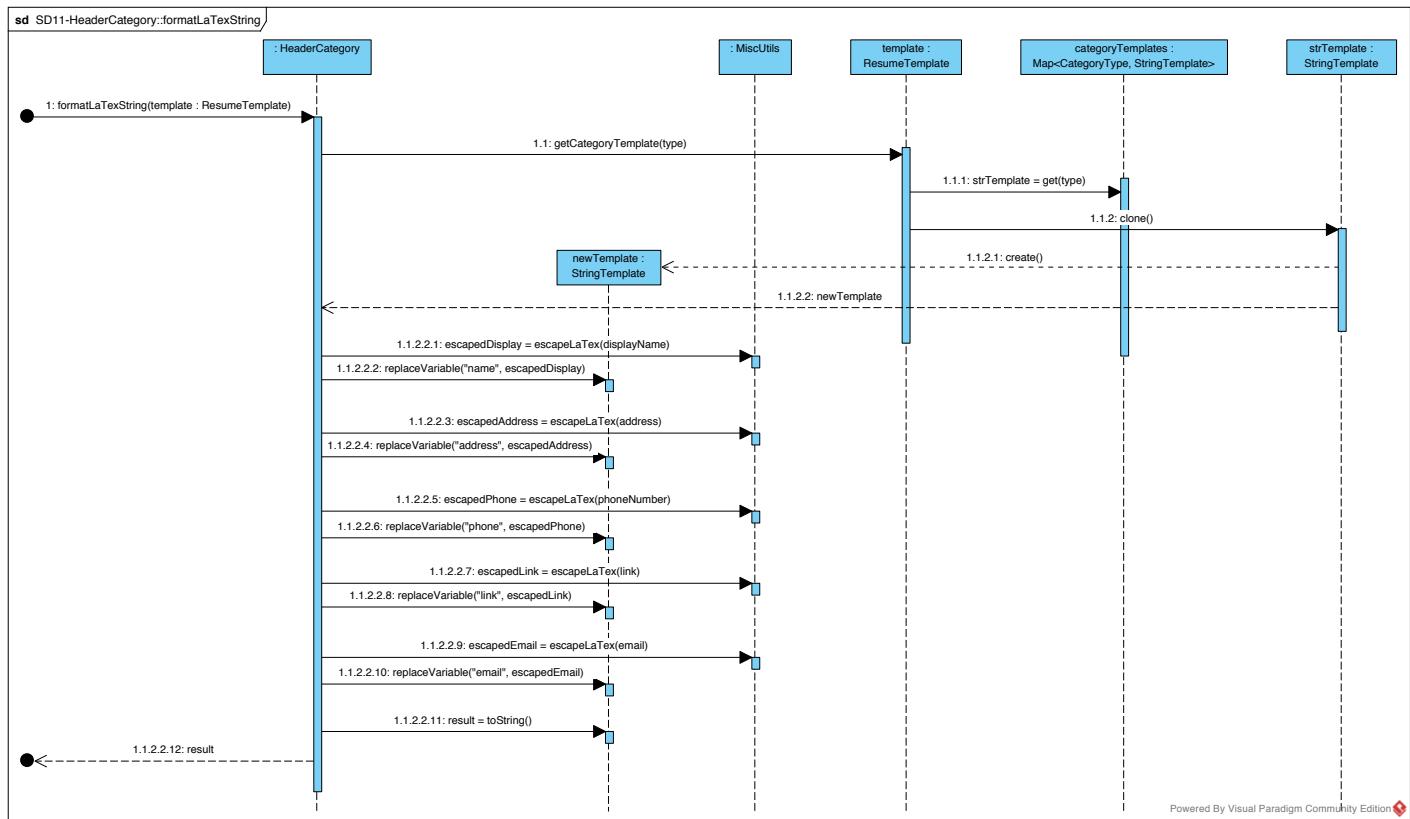


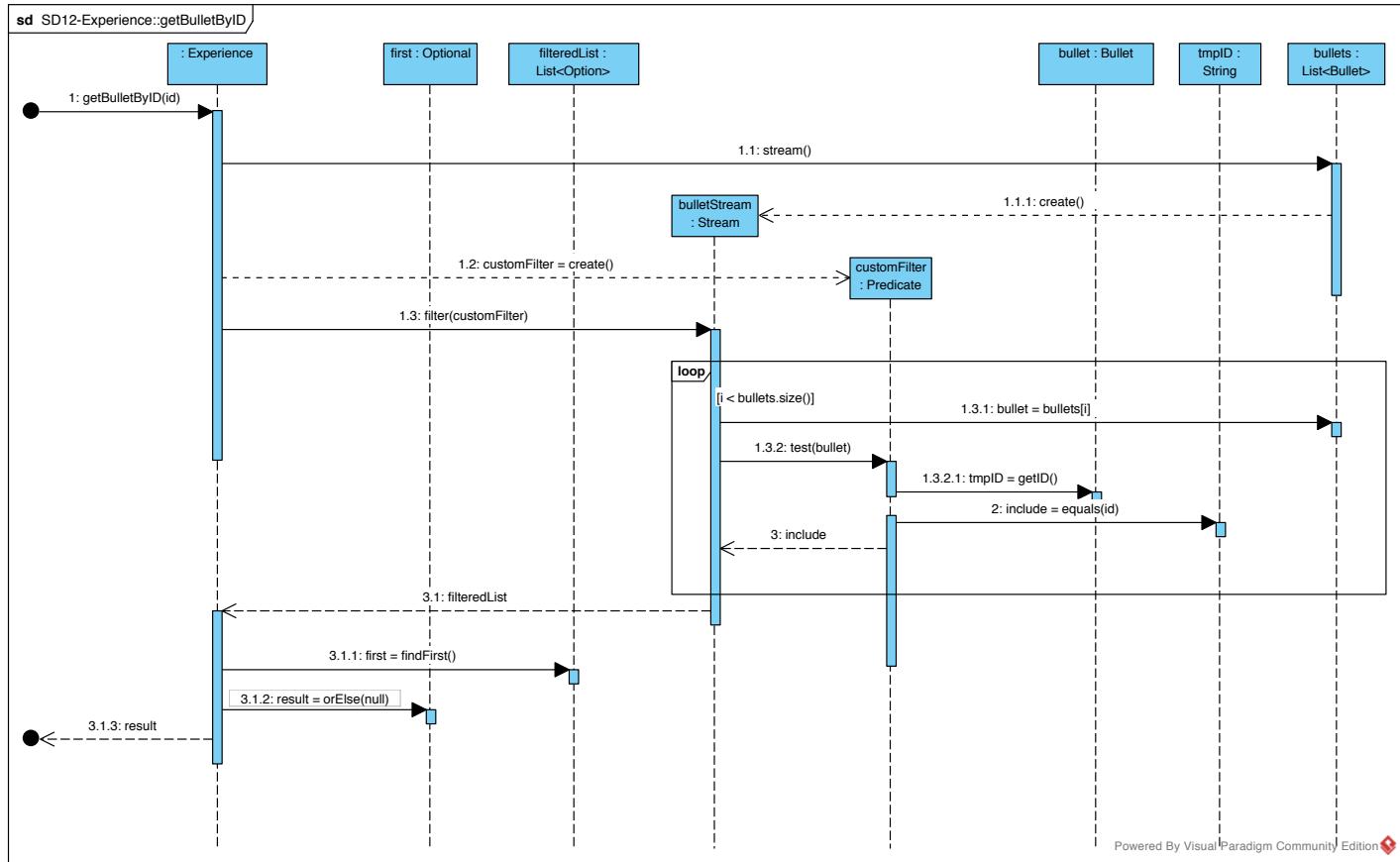


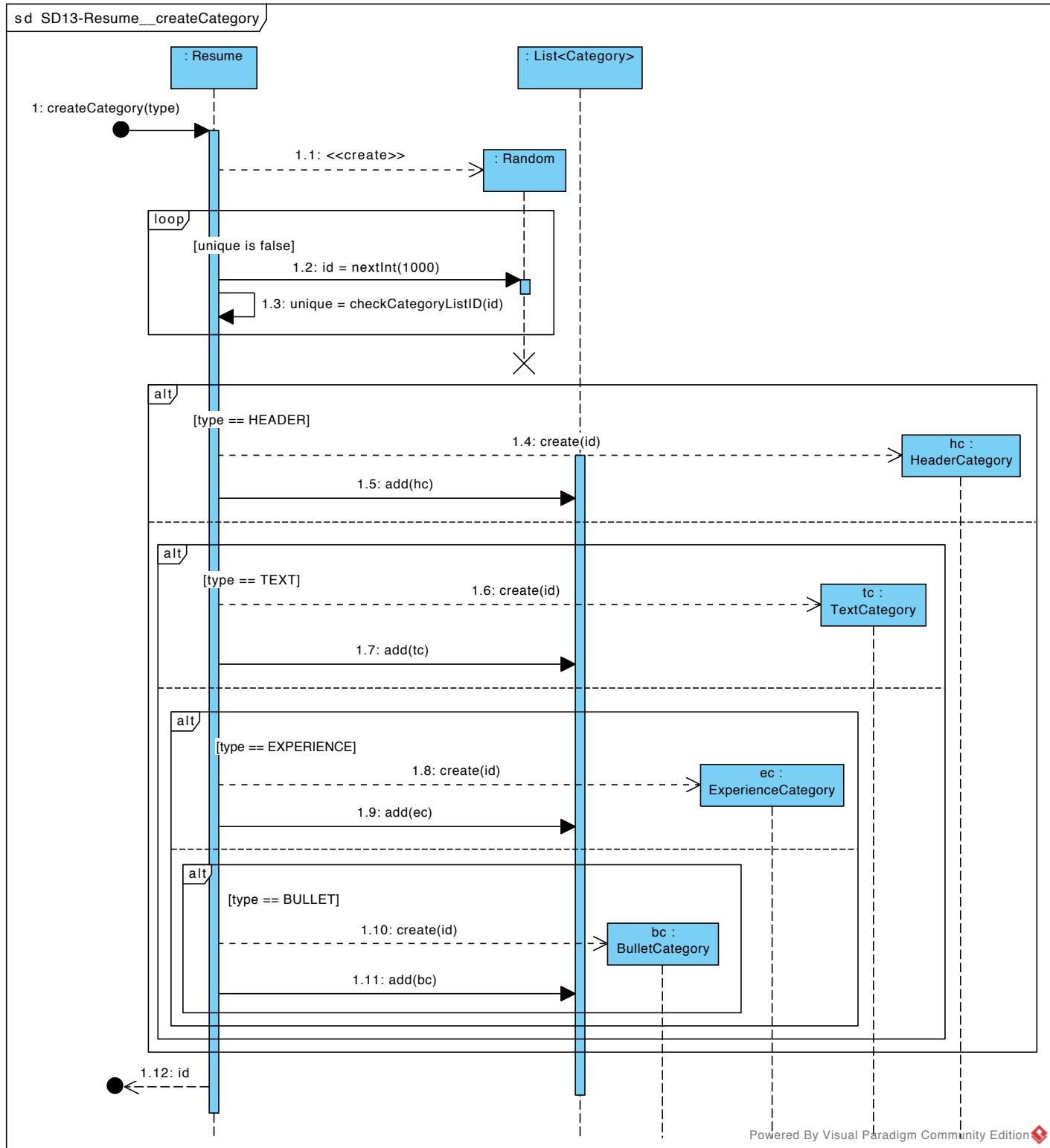


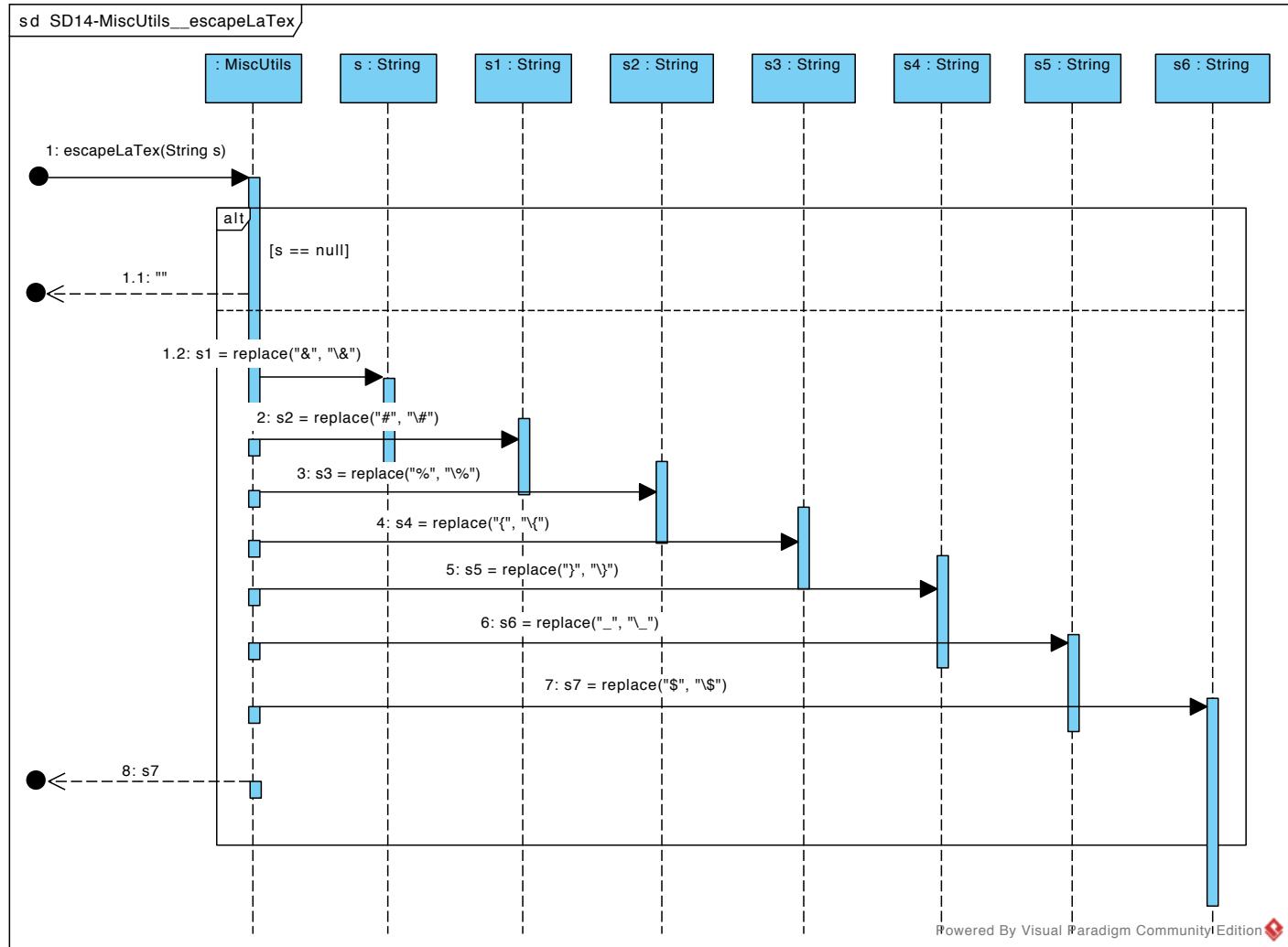


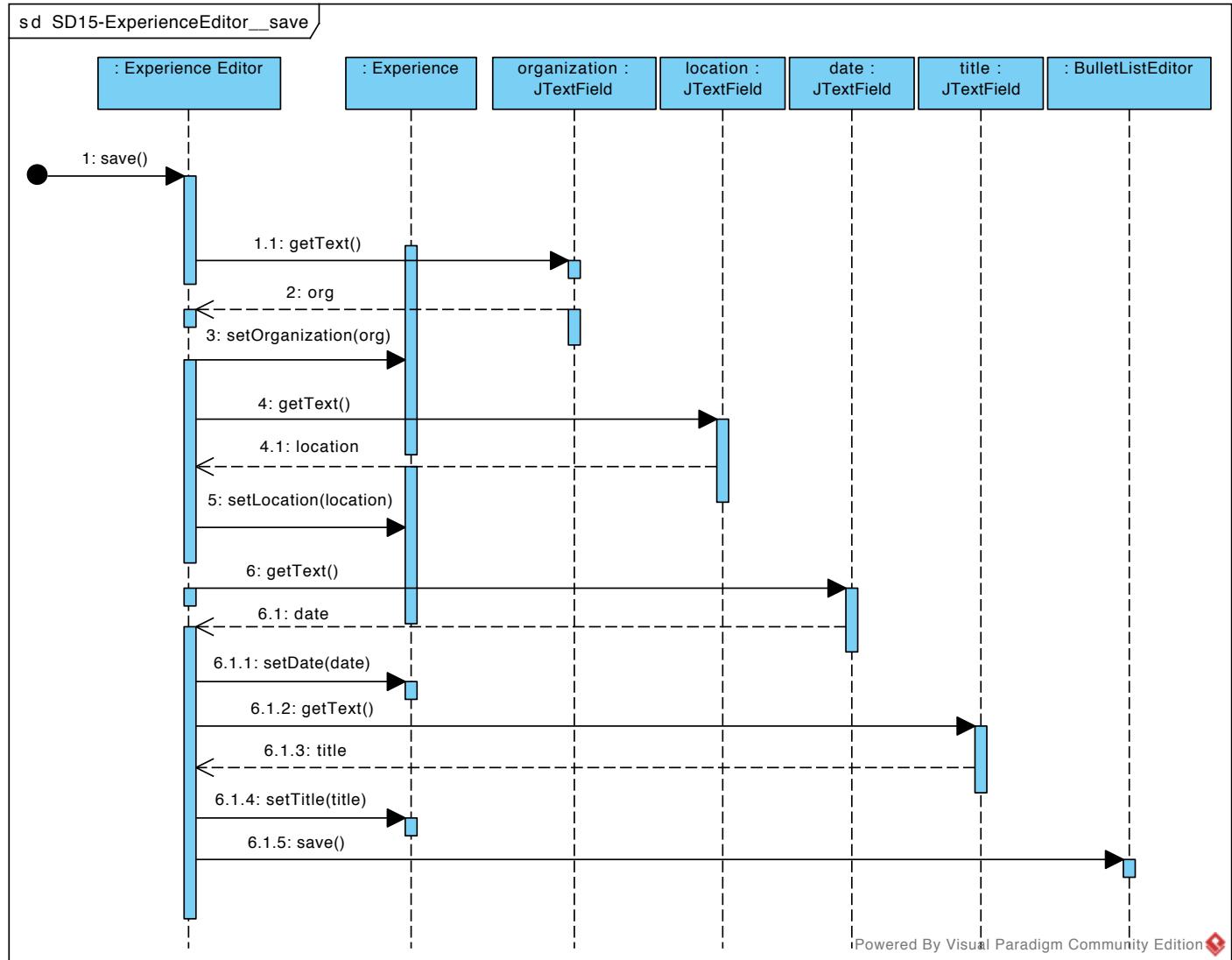












GRASP

Polymorphism:

- EditPanes
- Bullet/Experience
- IBulletContainer

Pure Fabrication:

- BulletListEditor
- IBulletContainer

High Cohesion:

- CategoryEditPanes
- Bullet/Experience
- ResumeExporter
- ResumeFileManager

Low Coupling:

- ResumeExporter
- HeaderCategoryEditPane
- ResumeFileManager
- EditorCategoryHeader

Controller:

- EditorController
- MenuController
- EditorMenuBar

Information Expert:

- EditorController
- HeaderCategoryEditPane
- EditorCategoryHeader

Indirection:

- MenuController

IMPLEMENTATION

DESIGN PATTERNS

Singleton

- Example: ApplicationConfiguration
- Justification: ApplicationConfiguration, as its name implies, handles configuration data for the program. Having multiple, desynced copies of this information would be irritating at best (the application randomly switching behavior) or dangerous at worst (the export temporary directory switching at random and thus causing an export to fail). This fits the problem statement for a Singleton, making this design pattern the best solution.

Facade

- Example: PDFFacade
- Justification: Making calls to the Spring server for the app requires handling multiple variables. To make these, the class needs to handle interacting with the custom BasicRequest class, GSON to parse file.io responses (when posting to a URL), and handling errors that arise within this subsystem. Since this is a large number of dependencies to add to another class attempting to contact the server, the Façade pattern was employed to lower coupling to these implementations, and also simplify the interface other classes must employ when communicating.

Visitor:

- Example: CategoryVisitor (concrete implementation: CategoryEditVisitor)
- Justification: The creation of CategoryEditPanes must be done based off of the type of the category received, feeding the category in question to the CategoryEditPane to allow it to ready itself to edit the fields within it. This poses a problem, however, as the EditorStage which uses the resultant component (and indeed, the rest of the code) is unaware of the runtime type of the Category. Putting the method “getEditor” within a Category does not make sense, as this would couple the data layer to objects in the presentation layer. To avoid this unnecessary coupling, then, and to facilitate a dynamic cast, a CategoryVisitor was created to enable this dynamic cast and category creation, without the need for a massive switch statement or undesirable coupling.

Template Method:

- Example: BasicRequest (to.us.resume_builder.business.server_connect.request)
- Justification: This was originally planned for more than it was used for. Originally, there was considered the possibility of connecting with the Spring server to facilitate users posting their PDFs to our server, necessitating a wide range of HTTP requests to be sent and received—thus, necessitating a framework for sending these requests. Thus, a Template Method was created in the form of BasicRequest::sendRequest to facilitate this. However, the user system was removed after deciding to outsource storing of files (as it

would have served no purpose at that point), leaving us with only two necessary HTTP requests: a POST request to the server (for compiling our PDF) and a separate POST to server.sh (for uploading the user's resume for a review).

Mediator:

- Example: EditorController
- Justification: The main editor interface is broken into two large components: the EditorStage (which holds the Editor itself), and the EditorCategorySelector, which allows the selection of the Category to edit. These are two discreet objects, each with a discrete job; however, they must receive and send events to each other. Direct coupling is not, however, desirable. Thus, the EditorController acts as controller to both of these classes, allowing them to forward important events (such as selecting a Category) without having to know or care where the events came from or go to.

Factory Method:

- Example: CategoryFactory
- Justification: Category classes require a dynamic creation methodology, depending on the type of category requested. This would necessitate a rather large switch statement, in addition to hard coding the types of available categories into the class. To circumvent this, the CategoryFactory was created, using a simple createCategory method, a map of possible Category types, and reflection to handle this creation dynamically.

Flyweight:

- Example: ImageCache
- Justification: Images are a memory-intensive resource which require no dynamic state, making the creation of several of the same image a pointless waste of resources. To prevent such waste, the ImageCache was introduced, implementing a Flyweight pattern to save all imported images and return the pre-existing copy when the same image is requested again.

ONLINE RESOURCES

Git Link:

- <https://github.com/ResuMaster>
- Includes access to the Client repository and the Spring server repository.

Ticketing System:

- <https://github.com/ResuMaster/CSI3471-Resume-Builder/issues>
- Includes a list of all issues on the Client repository.

Project Website:

- <http://resume-builder.us.to/>
- Includes access to each Iteration PDF and more information about the product.

Use Case Videos:

- <https://drive.google.com/drive/folders/1JcVPd9PTYjL0bT6TgNGubYtRfleKI6xx?usp=sharing>
- Updated information regarding iteration 1 use cases.

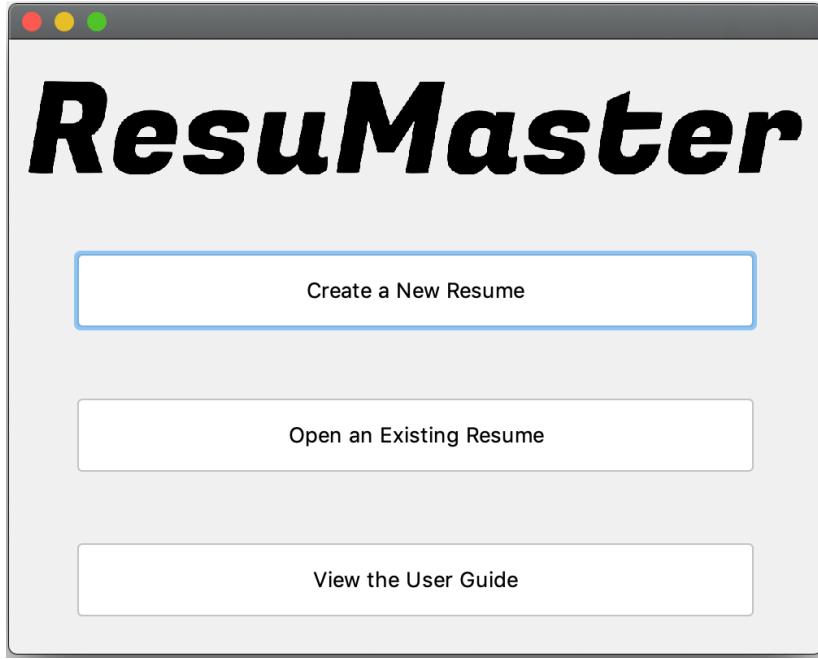
User Guide:

- <http://resume-builder.us.to/userguide.html>
- Links to a page with all information regarding usage of ResuMaster.

Iteration 3 Video:

- <https://docs.google.com/presentation/d/1TUotZWtotENJaWYSTbI63jPCZejTeLsXUYS9UHkHiqg/edit?usp=sharing>
- A link to a presentation used to present iteration 3 in the development of this software

GUI



Test

Export

Header
Profile
Education
Experience
Additional

Section Title: Matthew McCaskill Visibility: Delete Section

Link: www.linkedin.com/in/matthew-mccaskill

Email: me@matrmm.dev

Phone Number: (123) 456-7890

Address: Springfield, MO 12345

Add Section... Save

MyResume.pdf (1 page)

Billy Bob
www.myLinkedIn.com
1234 Nowhere Road, Far Away, Island, Earth
(637) 555-1234 • email@gmail.com

BIOGRAPHY
I am a _____ year student/employee at _____ University/Company. My goal is to _____ and I hope to be a (job title) at your company/university. I am skilled in _____ and _____. I have experience from my job at _____ doing _____.

EDUCATION
MY UNIVERSITY - City, State Month Year
Bachelor/Associate's Degree in _____

- This is what I did.
- This is what I accomplished.
- This is what I was involved in.
- These are some clubs I joined.
- These are the relevant courses I completed.
- Here is some more relevant information.

EXPERIENCE
WORKPLACE/COMPANY - City, State Month Year - Month Year
Worker

- These are my tasks.
- This is why it mattered.
- This is what I accomplished.

WORKPLACE/COMPANY - City, State Month Year-Month Year
Worker

- These are my tasks.
- This is what I learned.

SKILLS

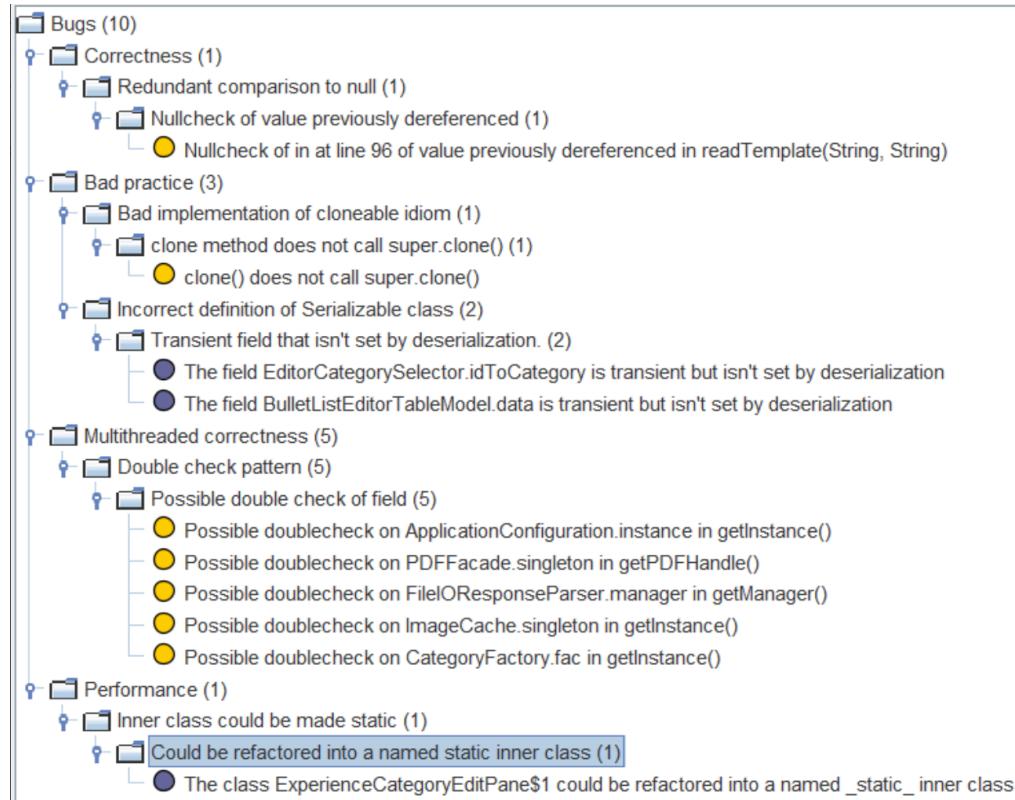
- Technical skills like Microsoft office or Windows 10.
- Social Skills like teamwork and leadership.
- Educational skills like teaching and learning.
- Unique to position (coding, cooking, driving abilities, etc.)

EVALUATION/TESTING

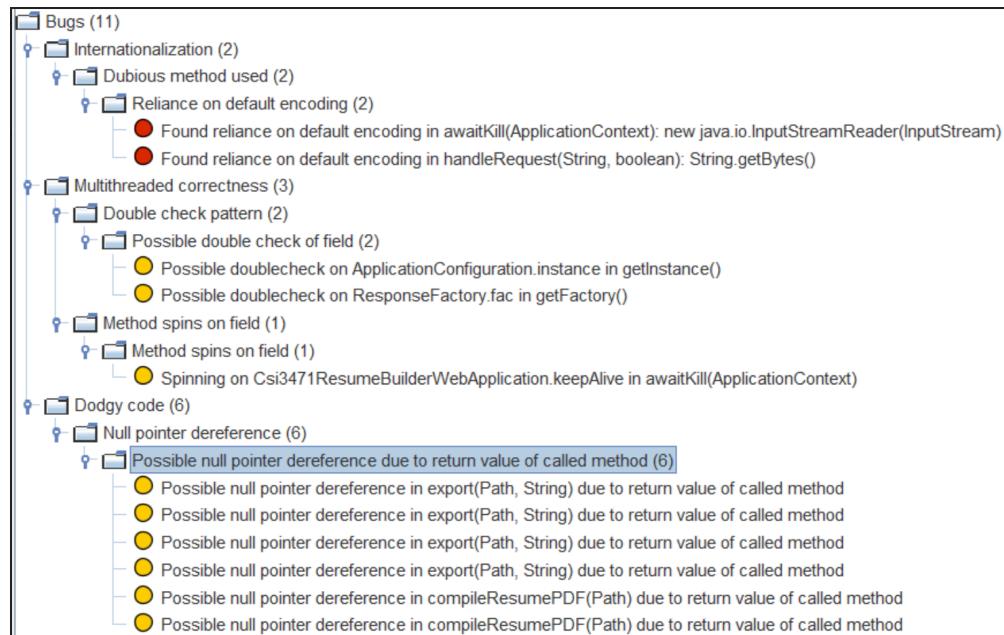
- Testing if saving a file and reloading it gives the same resume.
 - Testing each type of export and import that we offer, making sure data is stored and got correctly
- Package: **business**
 - Package: **controllers**
 - EditorController: testing add, remove, and changes in Categories
 - MenuController: voided
 - Package: **export_LaTeX**
 - ILaTeXConvertable: voided
 - ResumeExporter: null and non-null Resume tests
 - ResumeTemplate: voided
 - Package: **resume_file_management**
 - ResumeFileManager: exporting and importing various files
 - Package: **util**
 - MiscUtils: test with escape characters
 - StringTemplate: test clone and toString methods
- Package: **data** → test formatLaTeX String for all classes
 - Package: **resume_components**
 - Package: **category**
 - BulletCategory: test adding and removing Bullets
 - ExperienceCategory: test adding and removing Experiences
 - HeaderCategory
 - TextCategory
 - Resume: validate creating and removing Categories by checking for it in the Category list
- Package: **presentation**
 - Package: **category_edit_panes**
 - BulletCategoryEditPane: voided
 - ExperienceCategoryEditPane: voided
 - HeaderCategoryEditPane: voided
 - TextCategoryEditPane: voided
 - Package: **components**
 - BulletListEditor: voided
 - BulletListEditorTableModel: voided
 - ExperienceEditor: voided
 - EditorAddCategoryButton: voided
 - EditorCategoryHeader: voided
 - EditorCategorySelector: voided
 - EditorFrame: voided
 - EditorMenuBar: voided
 - EditorStage: voided

STATIC CODE ANALYSIS

SpotBugs Client-Side Results:



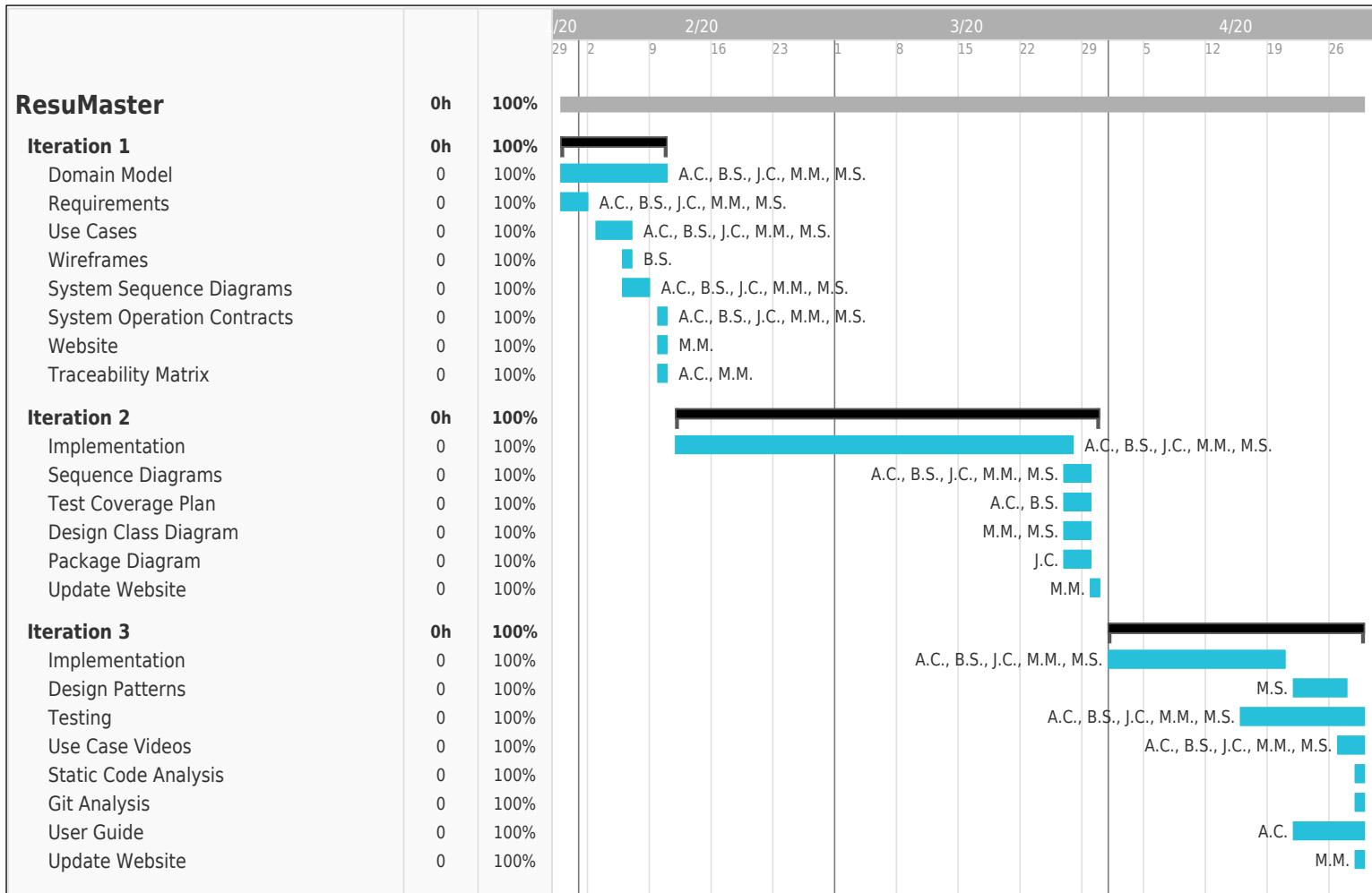
SpotBugs Server-Side Results:



**Extended information regarding these results can be found at

<https://drive.google.com/open?id=1EZq1EUFy6t-Fhg62JfB82b2TYpHGsXAx>

GANTT DIAGRAM

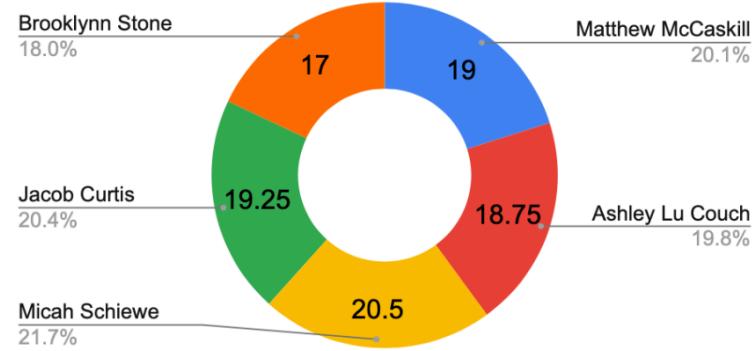


CONCLUSION

This software intends to make resumé building easier and more flexible for any job that you may be applying for. The initial vision was to have accounts for writers and reviewers so that feedback could be given on resumés, but in considering what was best for the users and knowing that all of the accounts would be on the same central device, this feature has been removed. Instead, we have developed a system for the user to store their resumé on a server with the ability to share that remote link with people over email. This removes the hassle for users to all be using the same device and instead allows them to back up their resumé online and use their own email client to communicate with others for feedback.

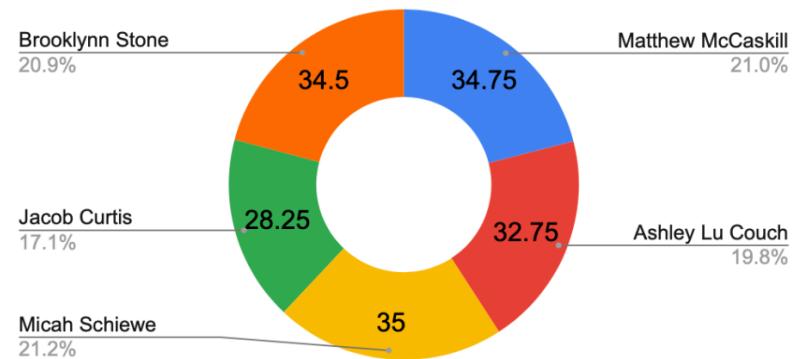
TIME SHEETS

Iteration 1



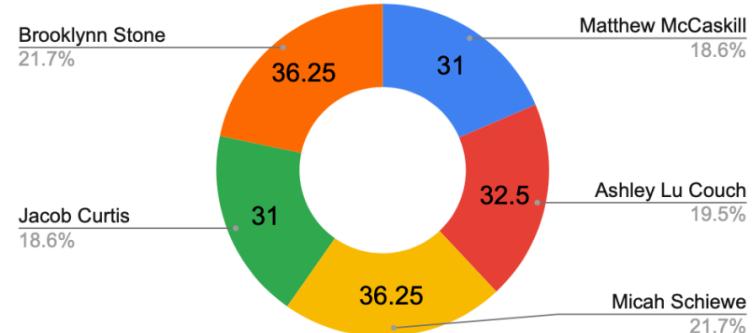
Total: 94.5 hours

Iteration 2



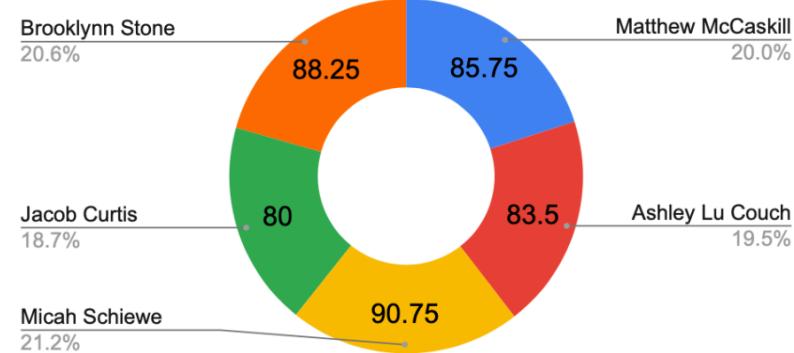
Total: 165.25

Iteration 3



Total: 159.5 hours

Overall



Total: 428.25

** Specific time card information can be found at https://drive.google.com/open?id=1p2C-AdhT8DfIXPUMw53yIKbXtdB_VQuVoh2YRvIKlZs