



**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA**

**ALAN VINICIUS CEZAR ENSINA**

**Estudo comparativo de tecnologias de processamento de  
linguagem natural para avaliação de histórias de usuário**

Florianópolis  
2022

ALAN VINICIUS CEZAR ENSINA

**Estudo comparativo de tecnologias de processamento de  
linguagem natural para avaliação de histórias de usuário**

Trabalho de conclusão de curso  
submetido ao curso de Sistemas  
de Informação para a obtenção  
do grau de Bacharel em  
Sistemas de Informação pela  
Universidade Federal de Santa  
Catarina – UFSC  
Orientadora: Prof<sup>a</sup> Dr<sup>a</sup> Fabiane  
Barreto Vavassori Benitti

Florianópolis  
2022

ALAN VINICIUS CEZAR ENSINA

## **Estudo comparativo de tecnologias de processamento de linguagem natural para avaliação de histórias de usuário**

Este Trabalho de conclusão de curso foi julgado aprovado para a obtenção do Título de “Bacharel em Sistemas de Informação” e aprovado em sua forma final pelo curso de Sistemas de Informação.

Florianópolis, XX de dezembro de 2022

---

Profº Drº. Álvaro Junio Pereira Franco  
Coordenador do Curso

**Banca examinadora:**

---

Profº Drª Fabiane Barreto Vavassori Benitti  
Orientadora

---

XXX

---

XXX

## RESUMO

Histórias de usuário são as representações das necessidades de um usuário e são utilizadas para facilitar o entendimento entre a equipe de negócios e a equipe de desenvolvimento para obter um maior acerto no desenvolvimento do produto com base na especificação. Porém, devido ao fato de serem escritas de maneira simples e curtas, diversas vezes podem causar dúvidas no momento da implementação. Sendo assim, é necessário encontrar uma forma de automatizar a avaliação dessas histórias afim de obter uma maior completude, uniformidade e consistência. Se tratando de automatização, o Processamento de Linguagem Natural (PLN) é uma subárea da inteligência artificial capaz de compreender automaticamente línguas humanas naturais capaz de automatizar diversos processos, porém devido ao alto número de tecnologias de PLN presente hoje no mercado, ainda é necessário compará-las para que seja possível aferir qual tecnologia possui, por exemplo, uma maior exatidão em seus processamentos, melhor performance e qual é a mais adequada no contexto de histórias de usuário. O presente trabalho pretende realizar uma análise comparativa entre soluções de PLN para a avaliação de histórias de usuário.

**Palavras-chave:** engenharia de software, histórias de usuário, processamento de linguagem natural, PLN

## **ABSTRACT**

User Stories are representations of a user's needs and are used to help the understanding between the business team and the development team to achieve greater accuracy in product development based on the specification. However, due to the fact that they are written in a simple and short way, they can often cause doubts at the time of implementation. Therefore, it is necessary to find a way to automate the evaluation of these stories in order to obtain greater completeness, uniformity and consistency. When it comes to automation, Natural Language Processing (NLP) is a subarea of artificial intelligence capable of automatically understanding natural human languages capable of automating several processes, but due to the high number of NLP technologies present on the market, it is still necessary to compare them so that it is possible to assess which technology has, for example, greater accuracy in its processing, better performance and which is the most appropriate in the context of user stories. The present work intends to carry out a comparative analysis between NLP solutions for the evaluation of user stories.

**Keywords:** software engineering, user stories, natural language processing, NLP

## LISTA DE FIGURAS

Figura 1 XXX ..... XX

## LISTA DE TABELAS

Tabela 1   XXX ..... XX

## **LISTA DE ABREVIATURAS E SIGLAS**

PLN	Processamento de Linguagem Natural .....	XX
-----	------------------------------------------	----



## SUMÁRIO

## 1. INTRODUÇÃO

“Tempo é dinheiro” (FRANKLIN, 1748) famosa frase dita por Benjamin Franklin na metade do século 18 ainda ecoa na cabeça de muitos seres humanos. Em busca de mais tempo as pessoas procuram então otimizar suas tarefas. Uma forma de otimizar as tarefas é a criação de automações. As automações buscam por uma melhor produtividade, redução de custos e maior tempo livre para se concentrar em outras tarefas que não podem ser automatizadas. Silva (2019) define que “... automação é um dos processos mais utilizados para a facilitação de inserção dos recursos tecnológicos. Através dessa tecnologia, são utilizadas ferramentas para soluções tecnológicas com o objetivo de otimizar e tornar simples os processos internos, além de diminuir custos operacionais.”

Um grande exemplo disso são as assistentes virtuais, como por exemplo a Alexa da Amazon, a Siri da Apple e o Google Home do Google. Esses assistentes virtuais são capazes de realizar diversas tarefas através de um simples comando de voz. Essa interação entre seres humanos e máquinas está cada vez mais presente nos sistemas, mas para que isso seja possível, é utilizado o Processamento de Linguagem Natural - PLN (RACKSPACE TECHNOLOGY, 2020).

Johnson (2021) define o PLN sendo um ramo dentro da Inteligência Artificial responsável em fazer com que as máquinas possam compreender a linguagem dos seres humanos, ou seja, o PLN funciona como um tradutor, permitindo assim que as tecnologias possam entender seus usuários, mesmo eles utilizando a linguagem natural.

O PLN também está presente em outras plataformas além das assistentes virtuais. Por exemplo, ele auxilia em sites de busca realizando interpretações entre o que o usuário digita com conteúdos de sites que poderão ser exibidos. Também está presente no auto-completar em plataformas de busca, onde sugestões automáticas são exibidas na tela no momento em que o usuário está digitando. Chatbots, que são utilizados por empresas para se comunicar com seus clientes, também fazem uso do PLN realizando a “tradução” do que o cliente deseja com possíveis soluções das quais as empresas podem oferecer (TAKE BLIP, 2019).

Para que seja possível criar sistemas voltados a automações, é necessário levantar os requisitos que esse sistema irá possuir. Em engenharia de requisitos, a etapa responsável para o levantamento dessas informações é a elicitacão. Para Thayer (1997), a elicitacão de requisitos é o processo em que os clientes e usuários são questionados pela equipe de desenvolvimento a falarem o quê espera como funcionalidades no sistema que será desenvolvido. Nessa etapa de elicitacão serão definidas as exigências, os recursos, os objetivos e as utilidades que o sistema deve cumprir.

Segundo Sommerville(2011, pág. 57):

Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições de seu funcionamento. Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma funcionalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações. O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é chamado engenharia de requisitos.

A especificacão de requisitos no desenvolvimento ágil pode ser feito por meio de histórias de usuário (User Stories). Através delas, o usuário utiliza de uma abordagem de escrever sobre os requisitos, tudo isso por meio de uma ou duas frases escritas na perspectiva de quem deseja o recurso/funcionalidade.

Para Cohn (2009, pág. 4), “uma história de usuário descreve a funcionalidade que será valiosa para um usuário ou comprador de um sistema ou software”. Já Rehkopf (2020) define histórias de usuário como “uma explicacão informal e geral sobre um recurso de software escrita a partir da perspectiva do usuário final. Seu objetivo é articular como um recurso de software pode gerar valor para o cliente.”

As técnicas de PLN também podem oferecer diversas vantagens para melhorar a qualidade das histórias de usuário. Segundo Raharjana, Siahaan e Fatichah (2021):

As técnicas de processamento de linguagem natural (PLN) oferecem vantagens potenciais para melhorar a qualidade das histórias de usuários. O PLN pode ser usado para analisar ou extrair os dados da história do usuário. Tem sido amplamente utilizado para ajudar no domínio da engenharia de software (por exemplo, gerenciamento de requisitos de software, extração de atores e ações no documento de requisitos, teste de software, etc.).

## 1.1 PROBLEMA

Cohn (2009) comenta que ao definir os requisitos de software a comunicação pode ser uma adversidade, pois aqueles que desejam um novo software devem se comunicar com quem irá desenvolvê-lo.

Heck (2014) propõe critérios específicos para avaliar a qualidade em histórias de usuários: completude, uniformidade, consistência e correção. Porém, muitos desses critérios, no entanto, requerem informações complementares que não são capturadas em um texto de história do usuário. Femmer (2013) define o termo *Requirement Smell* como indicador de má qualidade na especificação de requisitos. Femmer (2014) subdivide o *Requirement Smell* em 9 tipos: ambiguidade de advérbios e adjetivos, pronomes vagos, linguagem subjetiva, comparações, superlatividade, afirmações negativas, termos não verificados, *loopholes* (*brechas*) e referências não verificadas.

Dentro do contexto de histórias de usuário, seria possível avaliá-las utilizando soluções de PLN levando em consideração os critérios de qualidade?

Atualmente existem inúmeras soluções utilizadas para o PLN. Parker(2019) em seu artigo cita 12 ferramentas open source em diversas linguagens de programação, como por exemplo Python, Node e Java. Dentre as soluções citadas por Parker (2019), destaca-se a Natural Language Toolkit (NLTK) em Python, por ser a solução com mais recursos disponíveis, capaz de implementar todos os componentes de PLN e oferece suporte a vários idiomas. Outra solução que se destaca é a OpenNLP em Java. É hospedada pela Apache Foundation, ou seja, é fácil integrá-la com outros serviços da Apache. Assim como a NLTK, oferece suporte a vários idiomas e cobre todos os componentes de PLN.

No entanto, ainda é necessário compará-las para que seja possível aferir qual tecnologia possui, por exemplo, uma maior exatidão em seus processamentos e qual possui a melhor performance. Sendo assim, levando em consideração os critérios de qualidade (completude, uniformidade e

consistência), qual a solução mais adequada para o PLN no contexto de histórias de usuário?

Neste sentido, o objetivo central de desse trabalho é realizar um estudo sistemático de comparação entre pequenas soluções utilizando PLN para avaliação de histórias de usuário para avaliação da qualidade de histórias de usuário nos idiomas português e inglês.

## **1.2 SOLUÇÃO PROPOSTA**

Com base no cenário atual, no qual existem diversas tecnologias voltadas para PLN, se faz necessário uma análise comparativa entre essas tecnologias afim de definir qual ou quais tecnologias são mais adequadas para a avaliação de histórias de usuário.

Para que isso seja possível, serão selecionadas algumas tecnologias para que sejam previamente avaliadas em aspectos como por exemplo: documentação, linguagem de programação, conteúdo disponível na internet a respeito da tecnologia (sites, fóruns e *threads* em redes sociais) e o uso atual no mercado. Após feito esse levantamento de dados, as tecnologias que mais se destacarem serão selecionadas como objetos de estudo e serão implementados protótipos voltados a avaliação de histórias de usuário.

Esses protótipos serão avaliados levando em consideração critérios de qualidade em requisitos de software, como por exemplo eficiência, acurácia e funcionalidades. Quanto as histórias de usuário, serão selecionados alguns critérios de qualidade, conforme literatura, para avaliação.

Para além dos critérios de qualidade dos requisitos, pretende-se também avaliar nos protótipos aspectos relacionados a eficiência no processamento para os idiomas inglês e português e também a produtividade da tecnologia.

## **1.3 OBJETIVOS**

### **1.3.1 OBJETIVO GERAL**

Desenvolver um estudo comparativo entre soluções de PLN com o propósito de avaliar qual ou quais tecnologias são mais adequadas para

analisar critérios de qualidade em requisitos de software descritos como história de usuário.

### 1.3.2 OBJETIVOS ESPECÍFICOS

- Analisar e avaliar soluções atuais no mercado, comparando-as dentro dos critérios prestabelecidos;
- Implementar dois protótipos voltados para a avaliação de histórias de usuário utilizando as duas soluções de PLN mais bem avaliadas;
- Avaliar os protótipos desenvolvidos

### 1.4 METODOLOGIA

Metodologia é a estrutura filosófica dentro da qual a pesquisa é conduzida ou a base sobre a qual a pesquisa se baseia (BROWN, 2006). Já O’Leary (2004) descreve a metodologia como a estrutura que está associada a um conjunto particular de suposições paradigmáticas usadas para conduzir a pesquisa.

Sendo assim, dado o contexto de metodologia, o estudo seguirá o modelo científico em camadas (Research Onion) de Saunders (2007), seguindo a forma transversal, indutiva e interpretativa. Seguirá um modelo multimétodo, com procedimento de pesquisa bibliográfica (GIL, 2010), estudo comparativo das tecnologias (FACHIN, 2001), design e prototipação (SOMMERVILLE, 2011) e Goal Question Metric (GQM) (BASILI, CALDIERA, ROMBACH, 1994).

<b>Etapa</b>	<b>Atividades</b>	<b>Métodos</b>	<b>Resultados</b>
<b>Etapa 1</b> - Síntese da fundamentação teórica	- Sintetizar contexto histórico de processamento de linguagem natural, de histórias de usuário e de critérios de qualidade	Pesquisa bibliográfica (GIL, 2010)	Fundamentação teórica

<b>Etapa 2</b> - Estudo comparativo	<ul style="list-style-type: none"> <li>- Pesquisar as tecnologias mais utilizadas</li> <li>- Análise das tecnologias conforme os requisitos preestabelecidos</li> <li>- Definir as duas tecnologias mais promissoras</li> </ul>	Estudo comparativo (FACHIN, 2001)	Análise comparativa de potenciais soluções
<b>Etapa 3</b> - Prototipação	<ul style="list-style-type: none"> <li>- Implementar protótipos das soluções A e B em português e inglês</li> </ul>	- Design e prototipação (SOMMERVILLE, 2011)	Protótipos das tecnologias selecionadas
<b>Etapa 4</b> - Avaliação comparativa	<ul style="list-style-type: none"> <li>- Avaliar e comparar os resultados obtidos das tecnologias A e B</li> </ul>	- GQM (BASILI et al., 1994)	Avaliação das tecnologias e tabela comparativa

Tabela 01 - Etapas e metodologias aplicadas

Nova página para cada novo capítulo

## 2. FUNDAMENTAÇÃO TEÓRICA

Um parágrafo introdutório apresentando o capítulo.

### 2.1 HISTÓRIAS DE USUÁRIO

Segundo Francino (2017), o conceito de histórias de usuário foi introduzido pela primeira vez em 1998 na XP (Extreme Programming) comparando-as com Casos de Uso. Com o aumento da popularidade da XP e do Scrum, as histórias de usuário se tornaram uma abordagem muito conhecida para a definição de requisitos.

Uma história de usuário pode ser descrita como uma frase curta e semiestruturada onde é capaz de ilustrar os requisitos de um software na perspectiva do usuário, ou seja, ele pode ser usada para identificar o desejo do usuário com relação ao produto (RAHARJANA, I. K, HARRIS. F, JUSTITIA. A, 2020).

et al.  
Wautelet, Heng, Kiv e Kolp (2017) definem que uma história de usuário consiste de quatro elementos:

- **Papel:** comportamento esperado do ator no contexto do problema
- **Objetivo:** condição desejada pelas partes interessadas
- **Tarefa:** obrigação específica que devem ser realizadas a fim de atingir os objetivos
- **Capacidade:** a habilidade dos atores em atingir as metas com base em certas condições ou eventos

Já para Cohn(2009), Para facilitar a escrita de histórias de usuário, ele sugere o seguinte template:

*“Como <tipo de usuário>, quero <algum objetivo> para que <algum motivo>”*

Exemplo: Faltando o exemplo

*“Como <tipo de usuário>, quero <algum objetivo> para que <algum motivo>”*

Cohn(2009) também afirma que mais importante do que escrever histórias de usuários é a discussão a respeito dela, sendo assim, sugere que elas devem ser escritas em pequenos papéis ou até mesmo em notas adesivas, para que sejam facilmente expostas em paredes ou murais para facilitar o planejamento e a discussão.

## 2.2 CRITÉRIOS DE QUALIDADE

Ao realizar as especificações de requisitos de software, o IEEE recomenda 8 características bases para qualidade de requisitos: correta, não ambígua, completa, consistente, classificada por importância/estabilidade, verificável, modificável e rastreável (Cite um mais recente... reveja as características... vou mandar link também no whatsapp... Veja seção 5.2.5 IEEE Computer Society, 1994). Porém, por se tratar de uma norma um tanto quanto genérica, sabe-se que as especificações dificilmente atendem a todos esses critérios (GLINZ, 2000). Será?? Vc concorda com isso?

Tendo em vista a superficialidade dos critérios definidos pelo IEEE, Heck e Zaidman (2014) desenvolveram um framework voltado a verificação de requisitos, neste framework, foram definidos três critérios qualidade para verificação de alto nível: completude, uniformidade, consistência e correção.

Esse framework é para requisitos ágeis, certo? Melhore o contexto.

vc detalha abaixo... então, não precisa citar aqui



**Completeness:** Todos os elementos necessários na Área de Produto devem estar presentes.

**Uniformidade:** O estilo dos elementos da Área de Produto deve ser padronizado. Consistência e correção??

**Conformidade.** Todos os elementos devem estar em conformidade com a propriedade objeto da certificação.

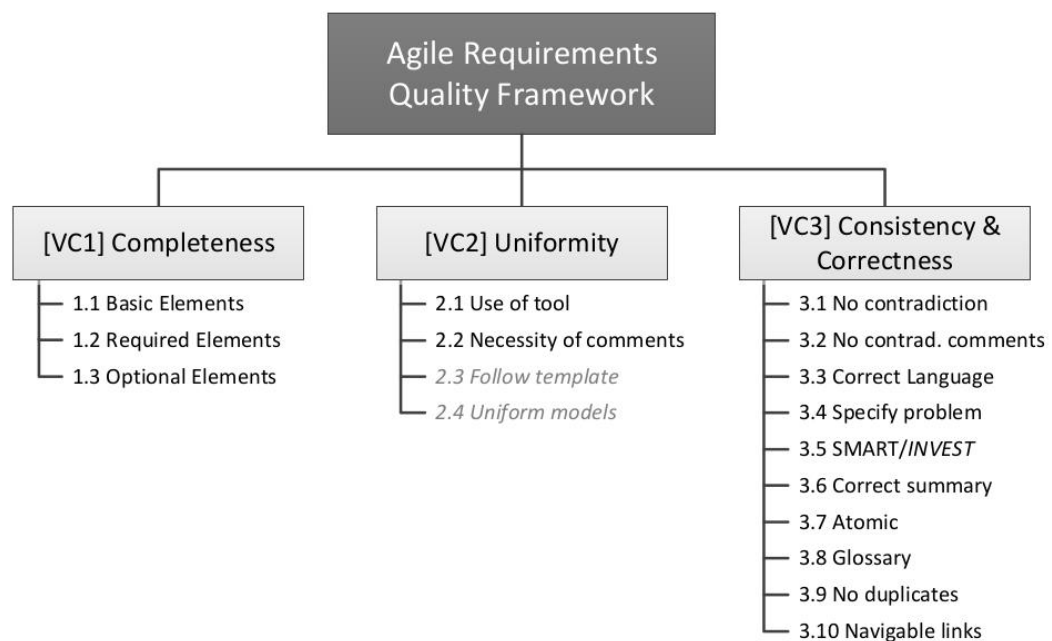


Figura 01: Critérios de qualidade definidos por Heck e Zaidman (2014)

Na Figura 01, é possível observar como cada critério de qualidade é subdividido, **onde** os itens em itálico são aplicados apenas para histórias de usuário, já o restante, são aplicados tanto para histórias de usuário quanto para solicitações de requisitos de software.

Faltando mencionar os artigos que eu te enviei (vou mandar novamente pelo Whats)...  
Sugiro vc descrever de forma ampla que há pesquisas nesta linha, e fazer uma tabela comparativa com os artigos encontrados x critérios

## 2.3 PROCESSAMENTO DE LINGUAGEM NATURAL

Em breve...

## X. REFERÊNCIAS BIBLIOGRÁFICAS

BARKER. D. **“12 open source tools for natural language processing”**, 2019. Disponível em:  
<https://opensource.com/article/19/3/natural-language-processing-tools> Acesso em 11 dez. 2021

BASILI, V. R.; CALDIERA, G.; ROMBACH, H. D. **“Goal Question Metric Paradigm”**. In: MARCINIAK Encyclopedia of Software Engineering. [S.l.]: John Wiley & Sons, 1994.

BROWN R. B, **“Doing Your Dissertation in Business and Management: The Reality of Research and Writing”**, 2006. Sage Publications

COHN, M. **“User stories applied for agile software development”**, 13. ed. Crawfordsville, Indiana. 2009. 263 p.

FEMMER, H. **“Reviewing Natural Language Requirements with Requirements Smells—A Research Proposal”**. Proceedings of IDoESE, 2013

FEMMER, H., FERNÁNDEZ, D.M., JUERGENS, E., KLOSE, M., ZIMMER, I., ZIMMER, J.: **“Rapid requirements checks with requirements smells: two case studies”**. In: Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering. pp.10–19. ACM, 2014

FRANCINO. Y. **“The essential guide to user story creation for agile leaders”**, TechBeacon, 2017. Disponível em:  
<https://techbeacon.com/app-dev-testing/essential-guide-user-story-creation-a-gile-leaders> Acesso em 17 mar. 2022

FRANKLIN. B. **“Advice to a Young Tradesman”**, 1748. Disponível em:  
<https://founders.archives.gov/documents/Franklin/01-03-02-0130> Acesso em 10 dez. 2021

GIL, A. C. **“Como elaborar projetos de pesquisa”**. São Paulo: Atlas, 2010. ISBN 5ª edição.

GLINZ, M. 2000, **“Improving the quality of requirements with scenarios”**. In: Proceedings of the World Congress on Software Quality (WCSQ), pp 55–60

HECK, P. KLABBERS, M. VAN EEKELEN, M. C. J. D. **“A software product certification model,”** Software Quality Journal, vol. 18, no. 1, pp. 37–55, 2010.

HECK, P. ZAIDMAN A., **“A quality framework for agile requirements: a practitioner’s perspective”**, 2014

HEATH, F. **The trouble with user stories**. 2020, DZone. Disponível em: <https://dzone.com/articles/the-trouble-with-user-stories-1> . Acesso em 04 dez. 2021

IEEE Computer Society (1994), **“IEEE recommended practice for software requirements specifications.”** IEEE Std 830-1993

O’LEARY Z. **“The essential guide to doing research”**, 2004. Sage.

RAHARJANA, I. K, HARRIS. F, JUSTITIA. A, **“Tool for generating behavior-driven development test-cases,”** J. Inf. Syst. Eng. Bus. Intell., vol. 6, no. 1, p. 27, Apr. 2020, doi: 10.20473/jisebi.6.1.27-36.

RAHARJANA, I. K, SIAHAAN. D e FATICHAH. C, **“User Stories and Natural Language Processing: A Systematic Literature Review,”** in IEEE Access, vol. 9, pp. 53811-53826, 2021, doi: 10.1109/ACCESS.2021.3070606.

RACKSPACE TECHNOLOGY, **“Dos chatbots à Alexa: a evolução do Processamento de Linguagem Natural”**, 2020. Disponível em: <https://www.rackspace.com/pt/solve/evolution-nlp> Acesso em 31 jan 2022.

REHKOPF. M. **“Histórias de usuários com exemplos e um template”**, 2020. Disponível em: <https://www.atlassian.com/br/agile/project-management/user-stories> . Acesso em 09 dez. 2021

SAUNDERS, M., LEWIS, P., & THORNHILL, A. **“Research Methods for Business Students”**, 2007, (6th ed.) London: Pearson.

SILVA, P. **“O que é automação e para que serve? Conversando com o CTO”**, 2019. Disponível em: <https://gobacklog.com/blog/o-que-e-automacao-e-para-que-serve/> . Acesso em 6 jan. 2022.

SOMMERVILLE, I. . **“Engenharia de software”**, 9. ed. Pearson. 2011. 529 p.

TAKE BLIP, **“Tudo sobre NLP: o que é processamento de linguagem natural e seus desafios na Inteligência Artificial”**, 2019. Disponível em: <https://www.take.net/blog/tecnologia/nlp-processamento-linguagem-natural/> . Acesso em 01 dez. 2021.

THAYER, R. H. e DORFMAN, M.; **“Introduction to Tutorial Software Requirements Enginnering”** in Software Requirements Engineering, IEEE-CS Press, Second Edition, 1997, p.p. 1-2.

WAKE, B. **“INVEST in Good Stories, and SMART Tasks”**, 2003. Disponível em: <https://xp123.com/articles/invest-in-good-stories-and-smart-tasks/> Acesso em 11 jan 2022.

WAUTELET. Y, HENG. S, KIV. S, KOLP. M, **“User-story driven development of multi-agent systems: A process fragment for agile methods,”** Comput. Lang., Syst. Struct., vol. 50, pp. 159–176, Dec. 2017, doi: 10.1016/j.cl.2017.06.007.