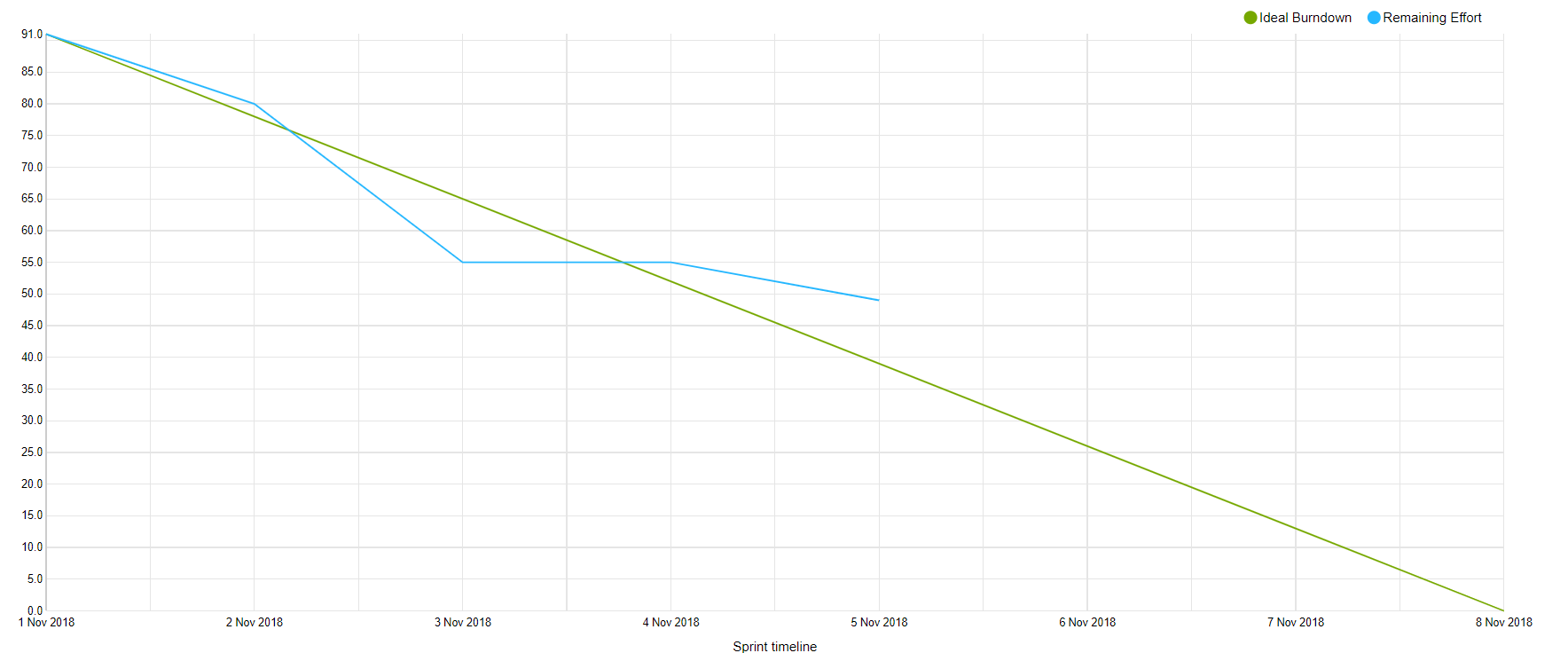
**WEEKLY LOG**

**(every Sunday)**

****

**Sprint 1 (4.11)**



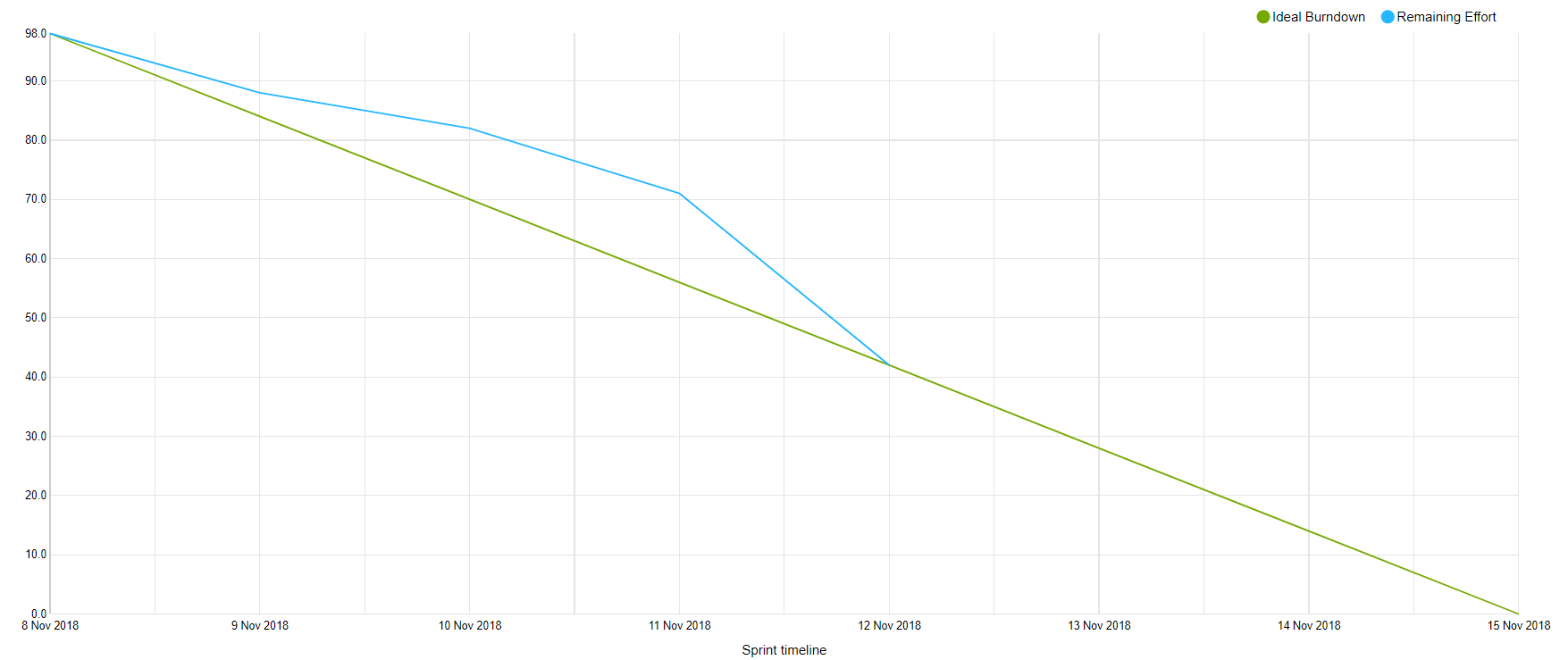
**What is done:**

* Database server
* Communication protocol (Search/Advanced search)
* Bookstore/Library implementation of model
* Activity diagram searching
* Class diagram : model, bookstore, library
* EER diagram : database book
* Database structure for books
* Domain and Logical model of book
* Search engine on database server

**What is in progress:**

* Class diagram : book service
* Book service redirecting and gathering book query
* GUI : search page (main page)
* GUI : showing results on web site
* Library/Bookstore connect to database
* Library expose REST for searching the book (just tests are missing)

**Sprint 2 (11.11)**



**What is done:**

* Class diagram : database
* GUI : showing result of search
* Use cases
* Database : availability
* Library CRUD
* Library CRUD test
* Class diagram (Library) updated
* Doc : add json examples to communication
* Database server : book details query
* Book service : book details
* Configuration of travis for C#
* Hibernate update for database schema

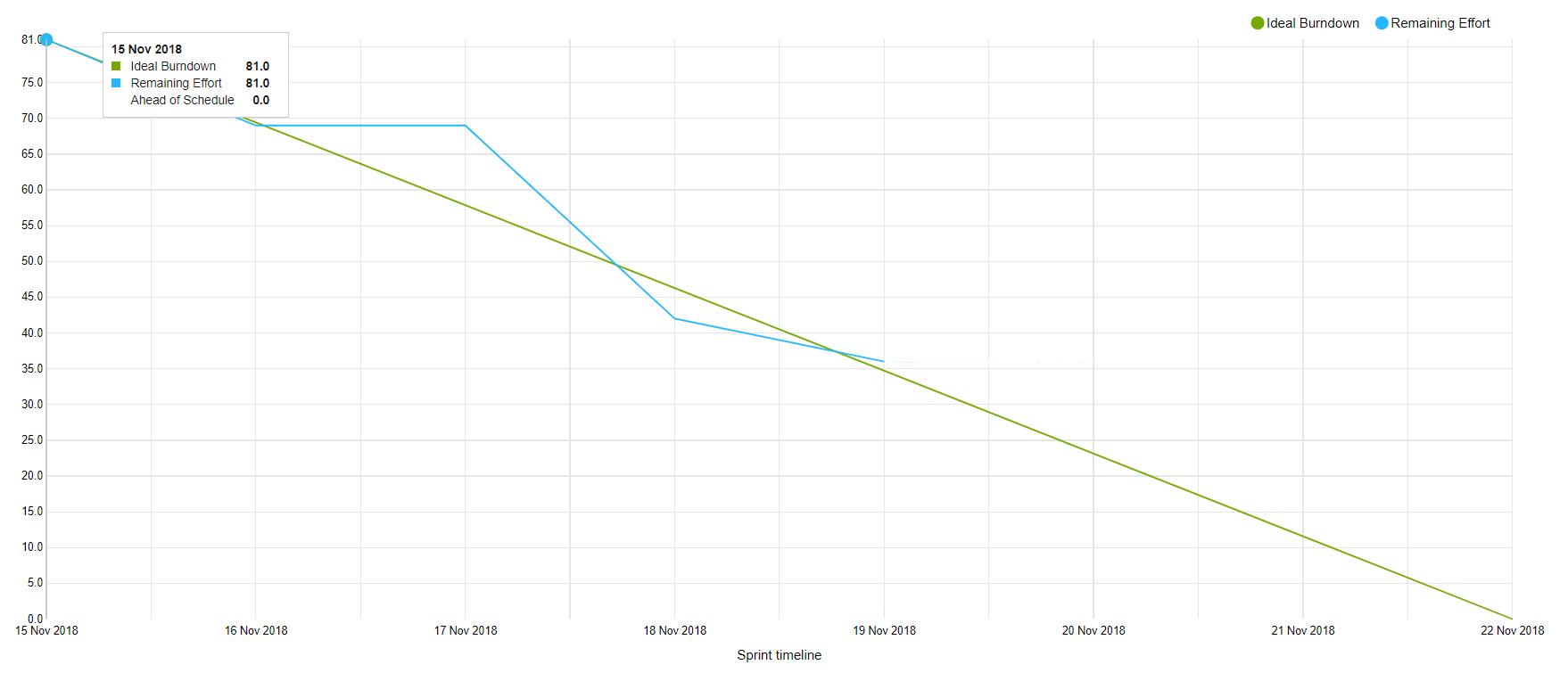
**What is in progress:**

* Bookstore CRUD
* Bookstore CRUD test
* Set up deploy
* GUI in progress

**New tasks:**

* Library database connection update
* Bookstore database connection update
* Database connection : handle details volumes
* Book service : handle details
* Library : create detailed book
* Library: book details -> read request
* Library: book details test

**Sprint 3 (18.11)**



**What is done:**

* Tests CRUD controller for bookstore
* Website : book details page
* Database : search and details for institutions -> communication
* Class diagram library updated
* Database : user tables + hibernate
* Database server : registration request
* Book service : registration communication with Database
* Use case diagram
* Database : checking book by ISBN
* Book service : handle registration request -> POST
* Library : documentation od API
* Library : change to new URI
* Book service : diagram updated
* Website : navbar

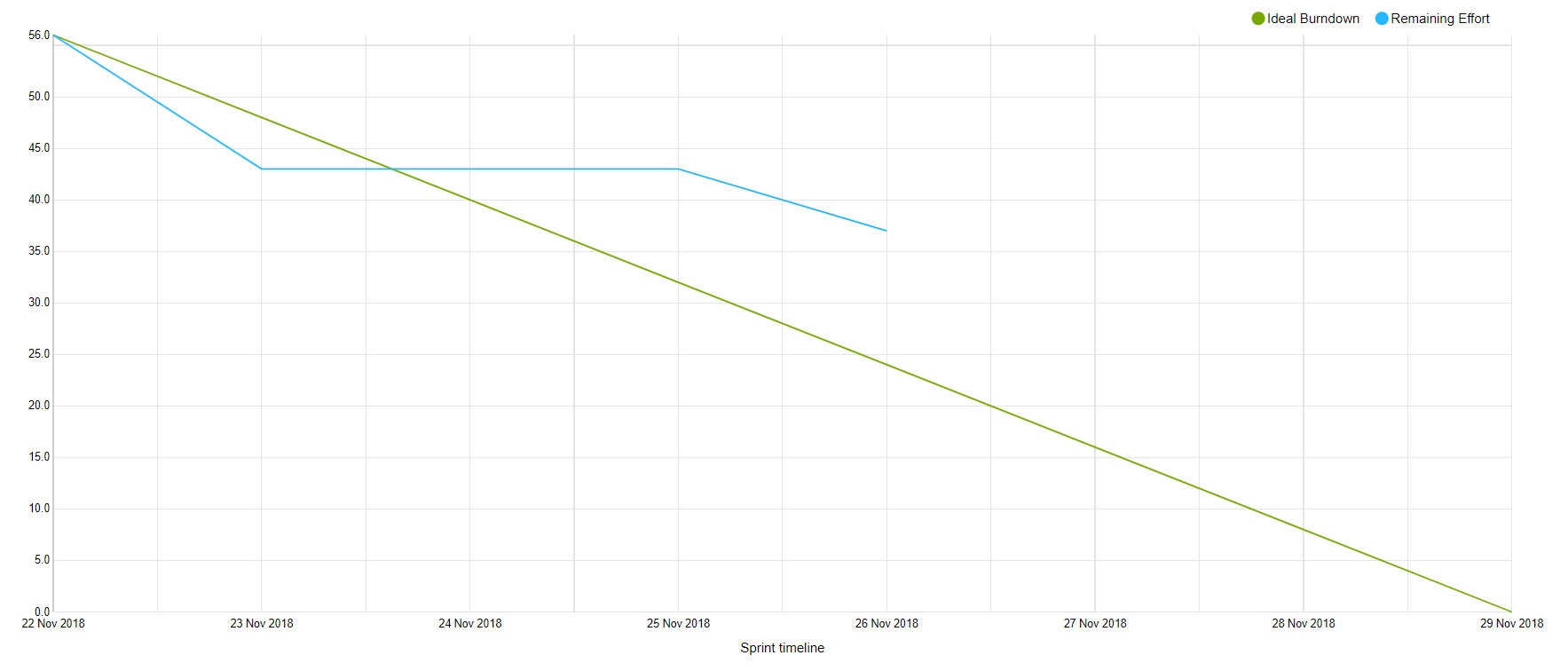
**What is in progress:**

* Website : admin page, registration page, connection with API
* Library : correct book details request
* Library : update documentation
* Database : tests for request handling
* Process report
* Bookstore : registration tests, controller
* Database documentation
* Web service documentation

**New tasks:**

* Database : library book details, bookstore book details
* Database : update connection of book details for institution, add institution name

**Sprint 4 (25.11)**



**What is done:**

* Error in database are fixed
* Use case
* Book service : request for borrowing the book from library, request for buying the book from the bookstore
* Database : order table
* Database : documentation updated
* Database : hibernate
* Database : add borrow order, buy order
* SSL
* Documentation of security

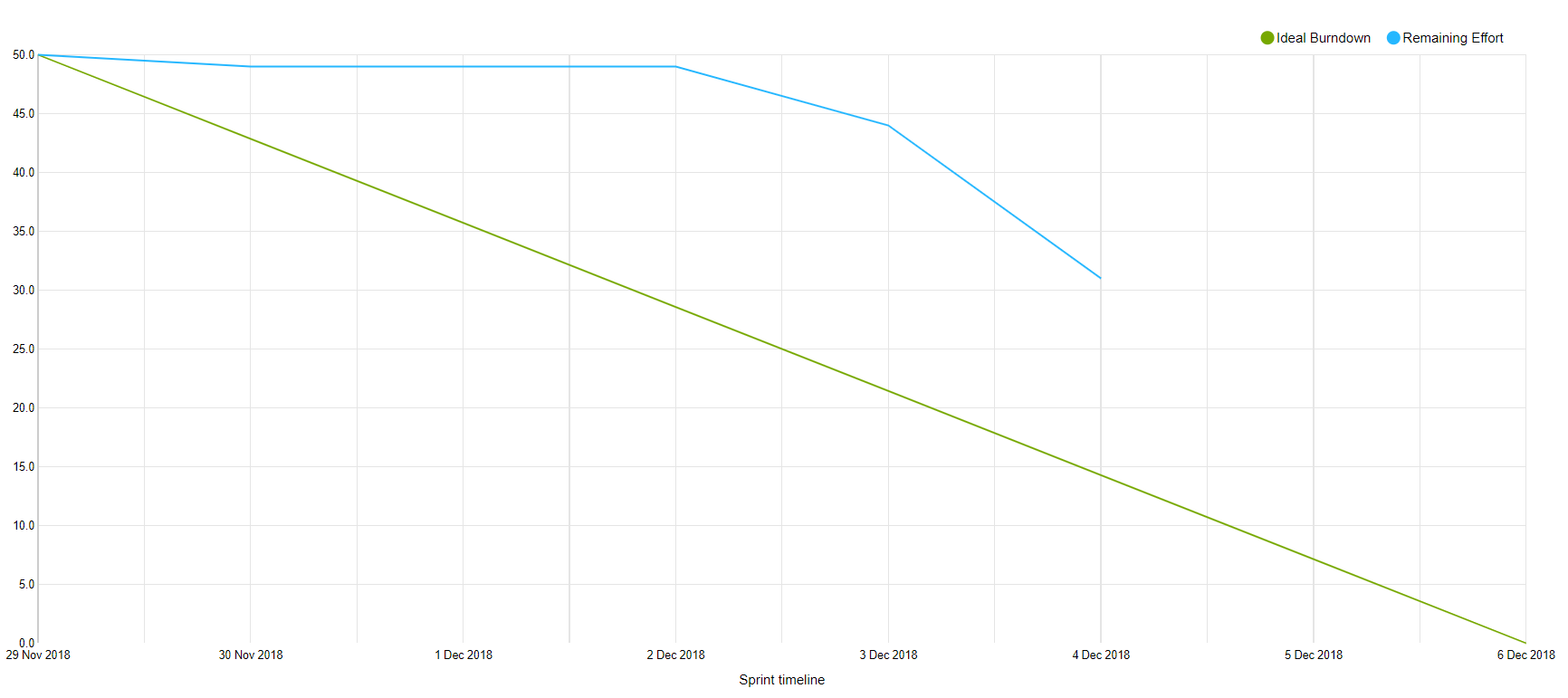
**What is in progress:**

* GUI : website -> connect both admins, list of libraries and books details for guest/customer, registration page connect form with API
* GUI : advanced search
* Fix search and advanced search
* Class diagram update
* Sequence diagram update
* Communication : borrow order, buy order
* Web service documentation update
* Hibernate query builder

**New tasks:**

* SSL
* Documentation of security
* Fix search

**Sprint 5 (3.12)**



**What is done:**

* Website : list of libraries and books in book details for guest / customer – layout
* Web services doc updated
* Find out how to make login
* Document login
* Database and Services add password
* Database add admins
* Sequence diagram login
* Web service : login layout
* Implement authentication

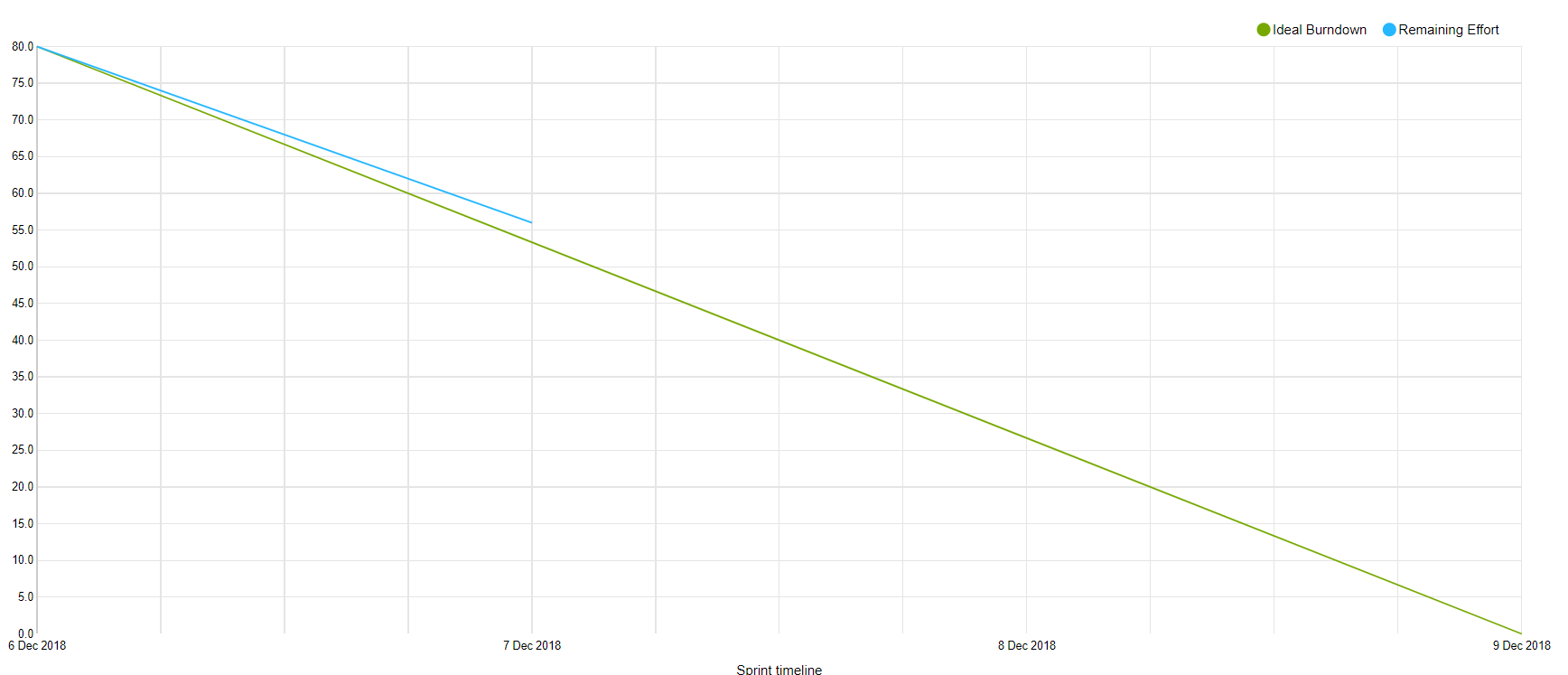
**What is in progress :**

* What is the progress : Website : connect both admins, connect book details list to API, registration page connect to API
* Website : storing cookies, adding cookie in request
* Website : changing layout on login
* Database documentation
* Change process report
* Document authentication
* Test library authentication
* Login encryption
* Add checking session key before request in Bookstore, Book service and Library

**DAILY LOG**

**Sprint 6**

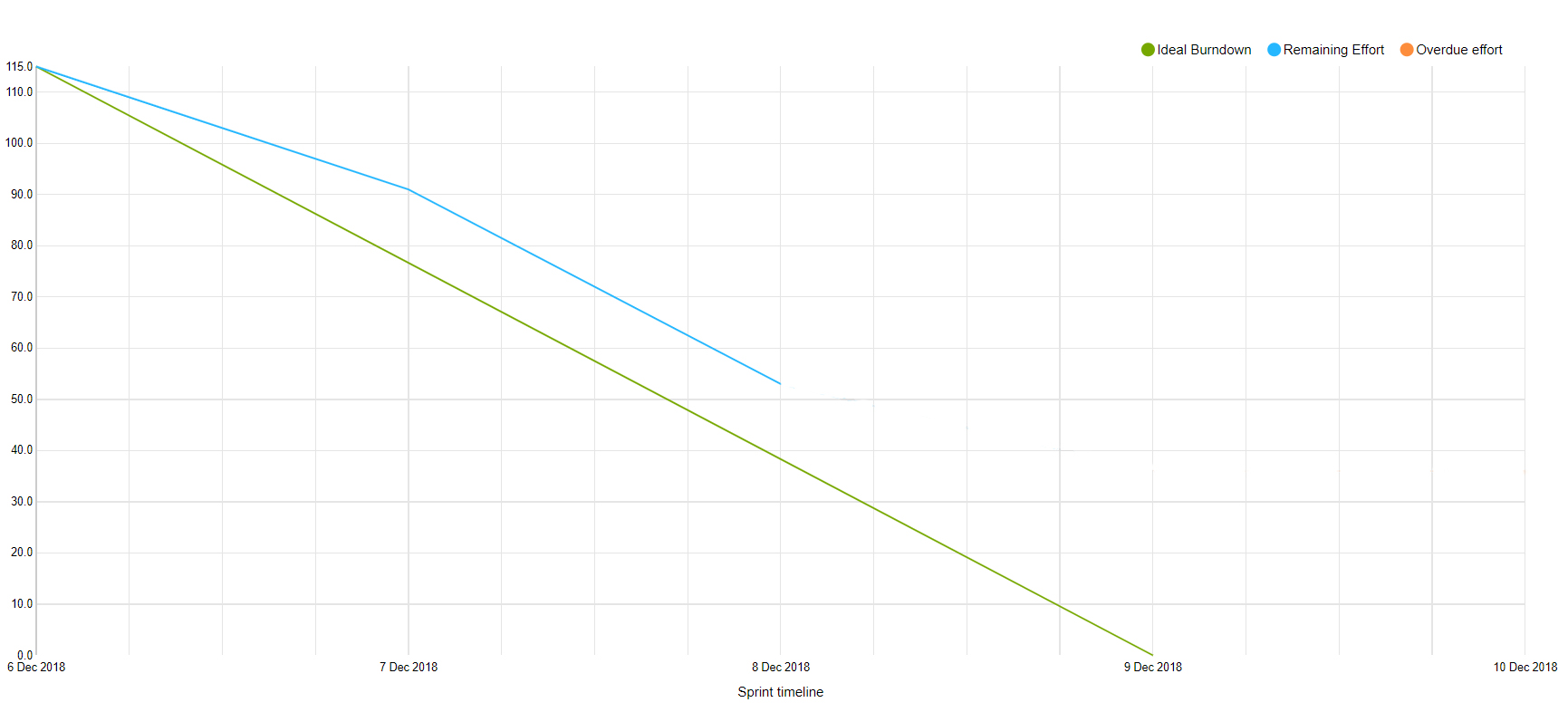
**6.12:**



**What is done:**

* website: registration page JS connect form with API
* document of authentication in book service and bookstore
* update library diagram
* add checking session key before request – library
* add checking session key before request – bookstore, book service
* database: handle confirm order for bookstore + communication
* use case

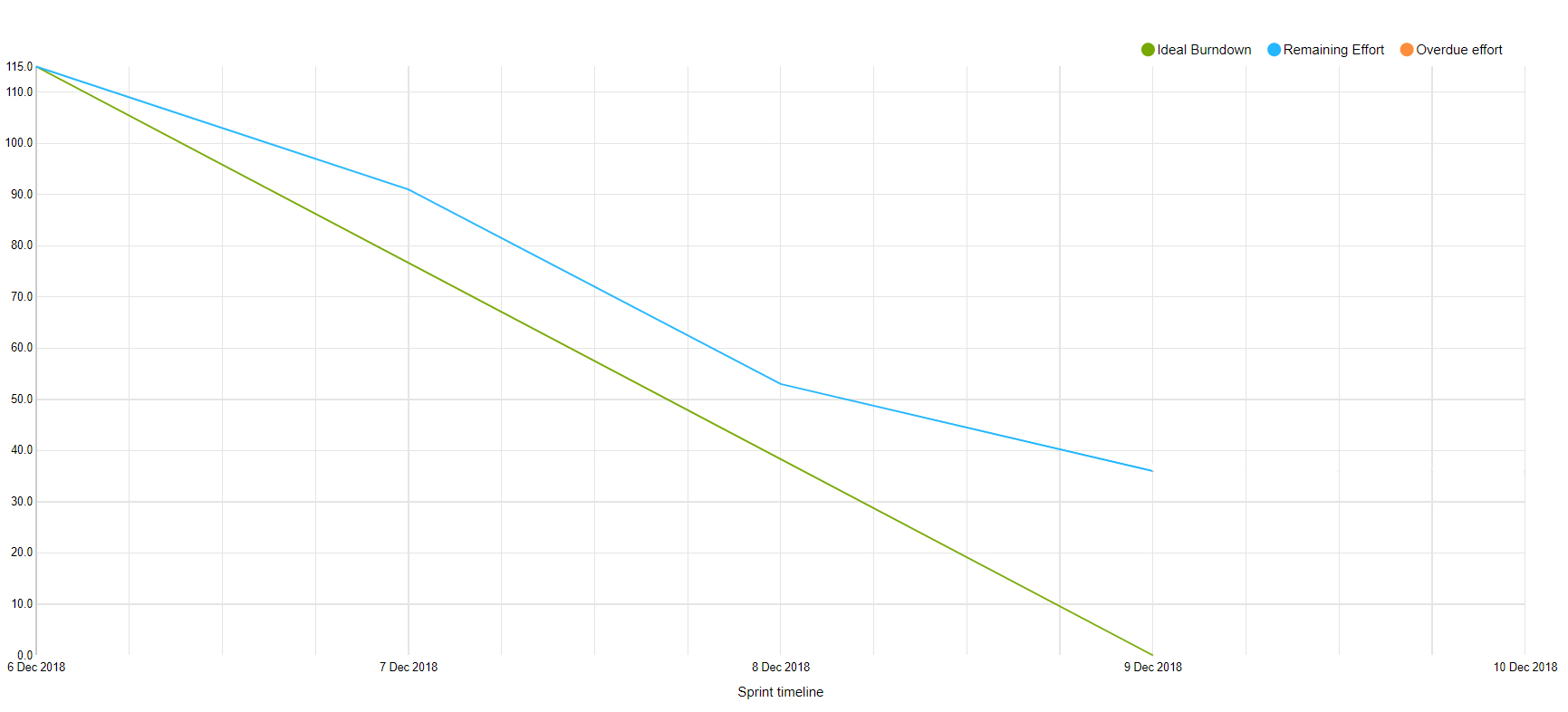
**7.12:**



**What is done:**

* login encryption
* database documentation
* database diagram updated
* authentication for bookstore document
* website : login storing cookies, adding cookie in request
* process report

**8.12:**

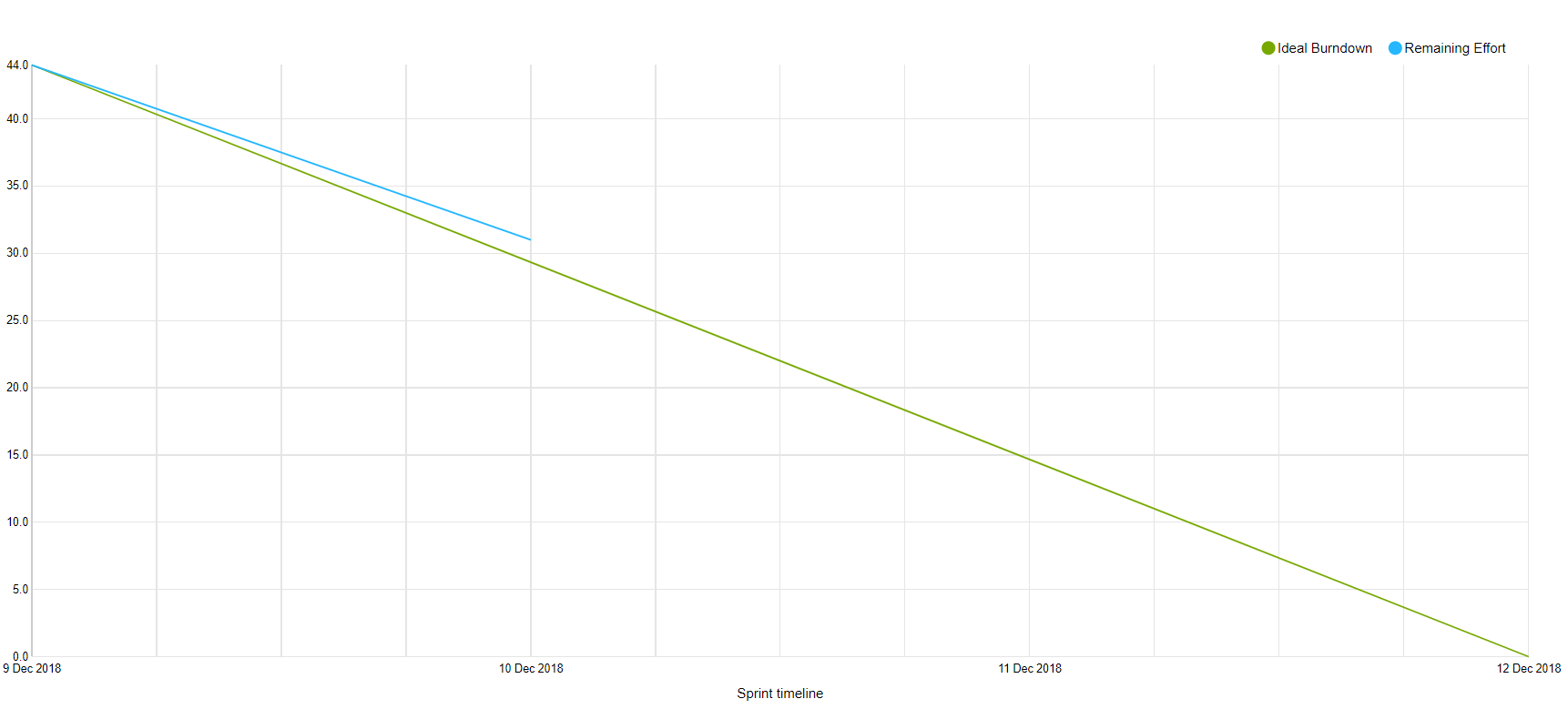


**What is done:**

* Database handle return book request + communication
* Update communication diagram
* Bookstore – get unconfirmed orders request
* Database handle get orders + communication
* Library authentication document
* Use case
* Library get orders – not returned book request
* Library return book request

**Sprint 7**

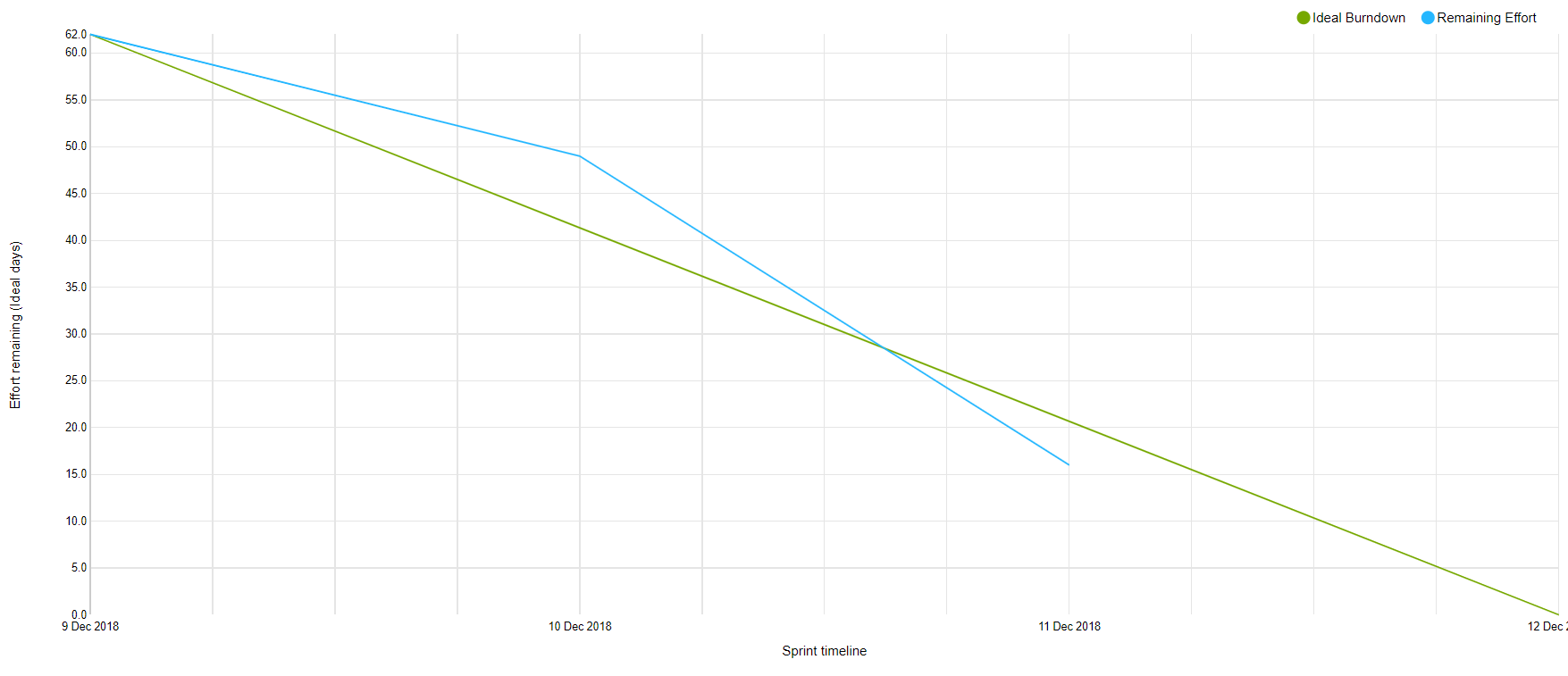
**9.12:**



**What is done :**

* Website : connect bookstore admin to API
* Website : navbar changing on login
* Website : connect advanced search to API
* Bookstore : confirm order manual test
* Library : return book manual test

**10.12:**



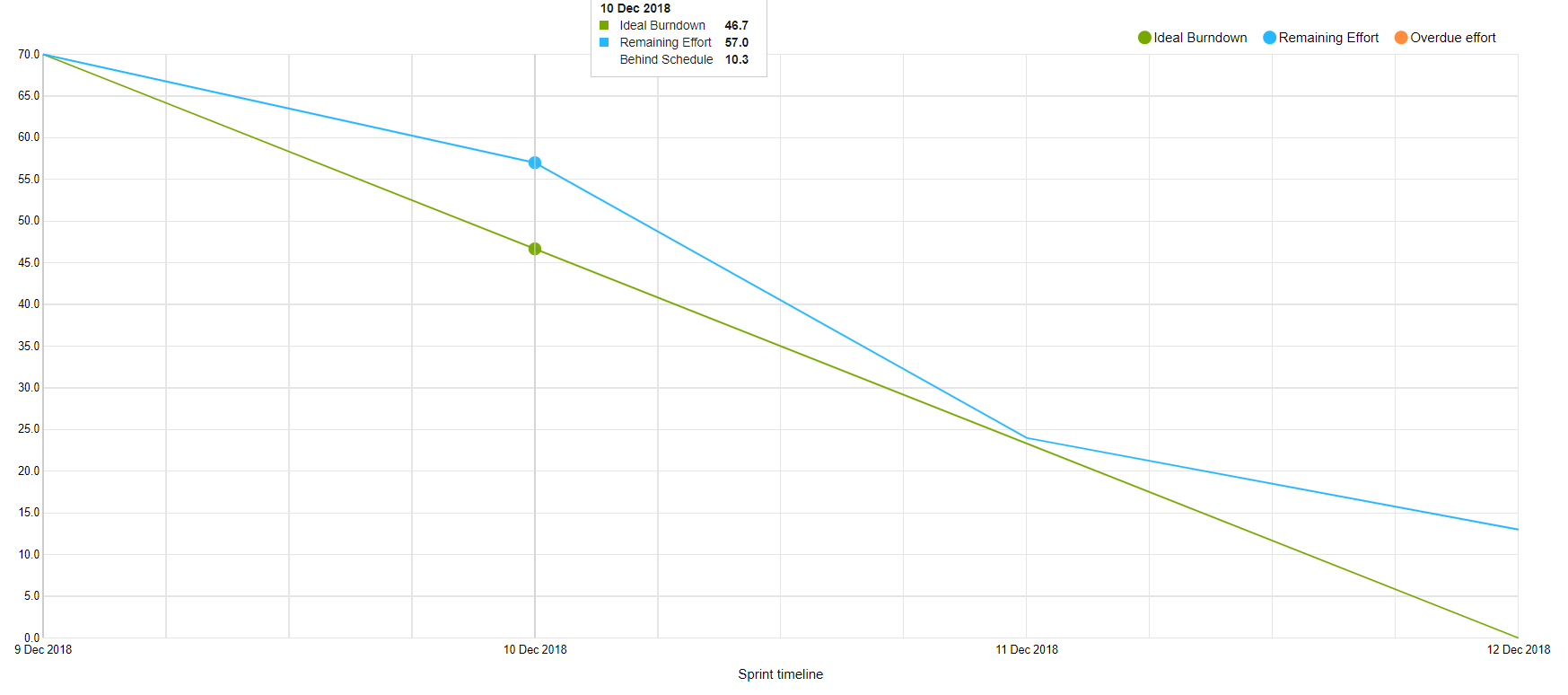
* Security documentation – analysis
* Website : library admin – layout for order
* Website : library admin orders connect to API
* Website : Advanced search corrected

**11.12:**

*\*We discover problem with the session key. When a session key is generated for the user, it’s not specifying for which type of user it belongs. In this way each user gain authorization for every service. In order to prevent it, each session key will store information of the user’s type.*

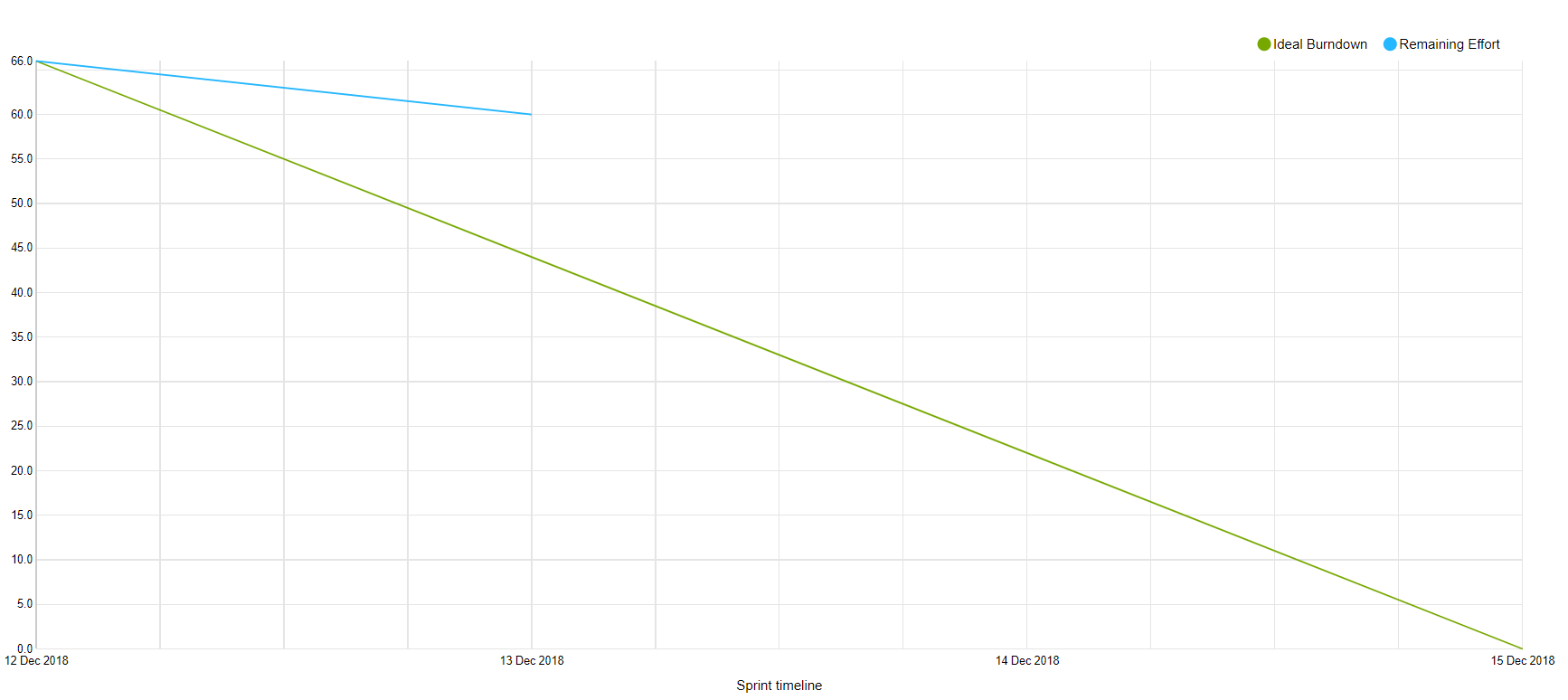
**New tasks :**

* Book service : add checking the type in session key
* Database : add URLs to admins
* Correct manage book use case



**Sprint 8**

**12.12:**



**What is done:**

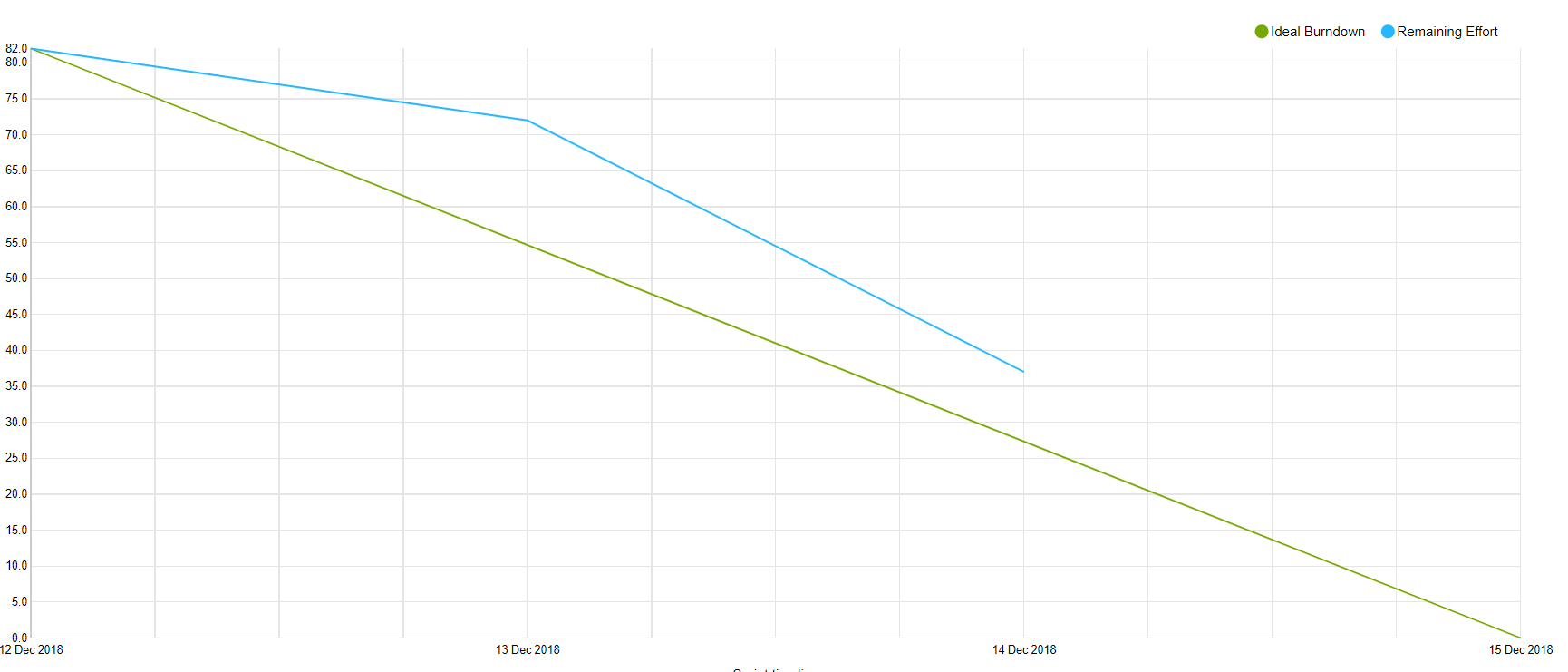
Project report : analysis – model

Library : logout request

Bookstore : logout request

Bookstore : logout request

**13.12:**



**What is done:**

Sending emails

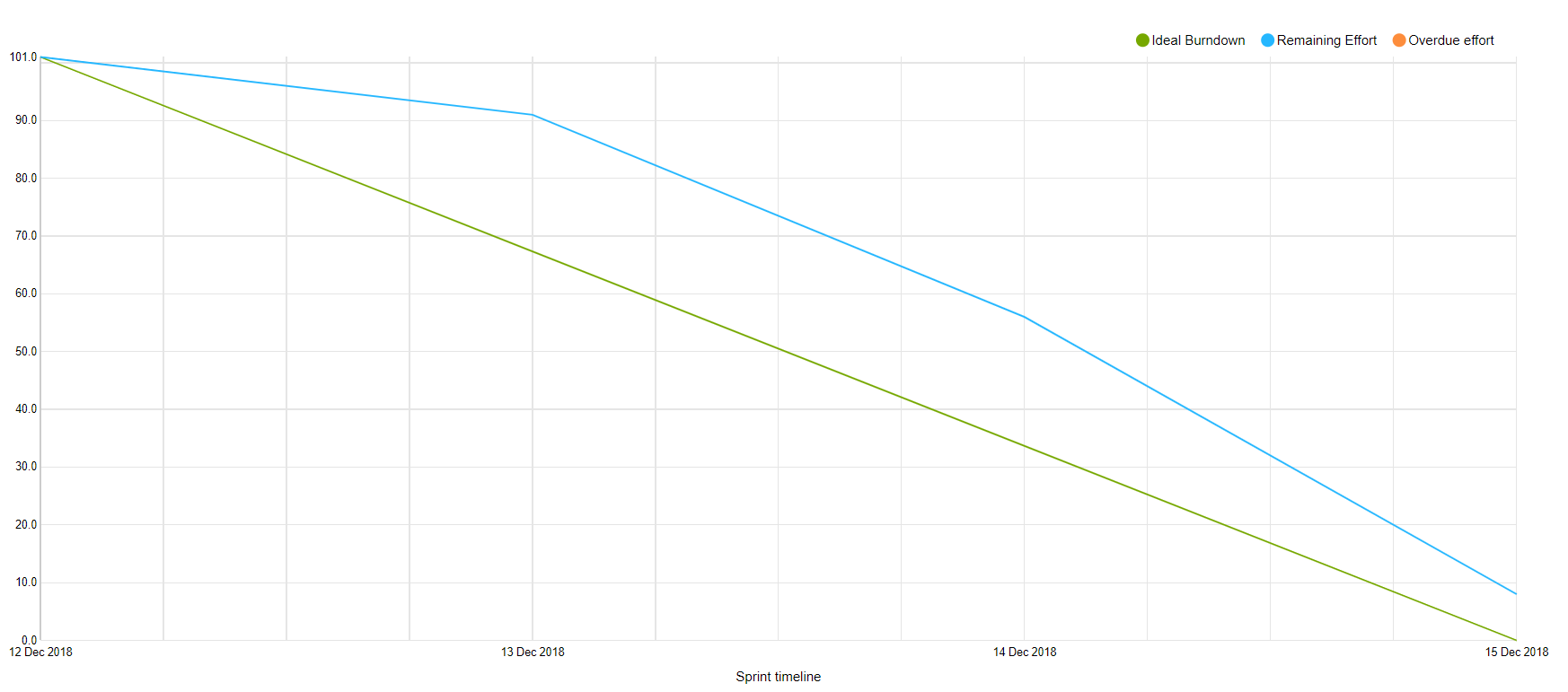
Website : showing borrow and buy button just when they are logged in

Website: add user id to post request of borrow and buy in book details

Project report : design communication, use cases, analysis model, architecture

Website : documentation of implementation, analysis, design

**14.12:**



**What is done:**

* Correct use cases
* Project report (analysis, tests, design, implementation, project future, conclusion)