

Sprint 1, Week 1 Writeup

This is the first week of the first sprint. The product backlog and sprint 1 backlog were moved into a new project space in Atlassian Jira, where the project will be managed from here on. In addition, [a GitHub repo was established](#) as the source code repository for the BottomLine application. The GitHub repo is publicly accessible at the link above while the Jira project is not public. Dr. Sangwan has been added to the Jira project.

Additional activities this week included refining of this sprint to decompose each requirement into smaller tasks, development of a testing strategy, and study of the required technologies needed to get the project off the ground. Development work was limited by the effort invested in administrative and project management tasks.

In keeping with the stated intent of the capstone course, the technologies chosen for this project are new and present a steep learning curve at this early stage of the project. It is hoped that this curve will flatten somewhat in future weeks as experience is gained in these areas.

Product Backlog

The table below shows a snapshot of the product backlog for the BottomLine app at the time of writing for this document. This is a living list and will change throughout the project. Each item on the backlog has been categorized broadly and prioritized.

The backlog does not show the requirements that have been assigned to a sprint, in this case Sprint 1. The requirements in Sprint 1 are shown later in this document and are no longer considered part of the general backlog.

The “live” version of the backlog lives in the Jira BottomLine project backlog, available here: <https://team-1622159142308.atlassian.net/jira/software/projects/BL/boards/1/backlog>

The product backlog has been expanded considerably from the prior week and has forty requirements as of this writing in excess of those already assigned to Sprint 1.

Issue key	Summary	Priority	Area	Requirement Type
BL-12	As a site user, I want to be able to update my contact information	High	Profile	functional
BL-13	As a site user, I want to be able to update my contact preferences	High	Profile	functional
BL-14	As a car dealer, I want to associate myself with particular car brands I sell	High	Profile	functional
BL-15	As a car dealer, I want to specify the location(s) of my store(s)	High	Profile	functional
BL-16	As a car dealer, I need to update any of the previously saved profile information	High	Profile	functional

BL-17	As a site visitor, I want to be able to use any modern web browser when viewing the site	High	Accessibility	non-functional
BL-18	As a site maintainer, I need to be able disable accounts	Medium	Admin	non-functional
BL-19	As a site maintainer, I need to be able enable disabled accounts	Medium	Admin	non-functional
BL-20	As a car shopper, I want to specify my geographic home area	High	Profile	functional
BL-21	As a car shopper, I need to be able to select options and colors specific to that make and model	High	Vehicle Config	functional
BL-22	As a car shopper, I want to see a running total of the MSRP for the specified make, model, and options I have chosen	High	Vehicle Config	functional
BL-23	As a car shopper, I need to be able to pick a vehicle make and model to configure	High	Vehicle Config	functional
BL-30	As a car shopper, I want to place a car configuration out to bid.	High	Shopper Bidding	functional
BL-31	As a car shopper, I only want to allow dealers within a certain geographic area to bid on my configuration because I am only willing to travel a certain distance to buy a new vehicle.	Medium	Shopper Bidding	functional
BL-32	As a car shopper, I want to be able to specify how long my configuration is available to bid on.	Low	Shopper Bidding	functional
BL-33	As a car shopper, I want to view bids on my configuration(s)	High	Shopper Bidding	functional
BL-34	As a car shopper, I want to receive automatic notifications of new bids on my biddable configuration(s)	High	Shopper Bidding	functional
BL-35	As a car shopper, I want to select a winning bid for my vehicle configuration.	High	Shopper Bidding	functional
BL-36	As a car shopper, I want to ask a question of a bidding dealer.	Medium	Shopper Bidding	functional
BL-37	As a car shopper, I want to view my vehicle configuration history.	Medium	Config History	functional
BL-38	As a car shopper, I want to modify a vehicle configuration.	Medium	Vehicle Config	functional
BL-39	As a car dealer, I want to view vehicle configurations available for bid.	High	Dealer Bidding	functional
BL-40	As a car dealer, I want to place a bid on a configuration.	High	Dealer Bidding	functional
BL-41	As a car dealer, I want to check the status of my bid(s).	High	Dealer Bidding	functional
BL-42	As a car dealer, I want to view my bid history.	Medium	Dealer Bidding	functional
BL-43	As a car dealer, I want to modify a bid.	High	Dealer Bidding	functional
BL-44	As a car dealer, I want to answer a question from a shopper.	Medium	Dealer Bidding	functional
BL-45	As a car dealer, I want to receive notifications for new configurations that I am eligible to bid on.	Medium	Dealer Bidding	functional

BL-46	As a site visitor, I would like I would like to be able to read or view an explainer about the BottomLine process for car shoppers	Low	Help	functional
BL-47	As a site visitor, I would like I would like to be able to read or view an explainer about the BottomLine process for car dealers	Low	Help	functional
BL-48	As a car shopper, I would like to see help in navigating the site	Medium	Help	functional
BL-49	As a car dealer, I would like help in navigating the dealer bid process on BottomLine	Medium	Help	functional
BL-50	As a car shopper, I need to be able leave feedback (review) vehicle dealers	Low	Feedback	functional
BL-51	As a site developer, I would like the site UI to make use of accepted usability standards (i.e. W3C) wherever possible so that it is available to as wide an audience as possible	High	Accessibility	non-functional
BL-52	As a site maintainer, I would like the site to be hosted in the cloud	Medium	Architecture	non-functional
BL-53	As a site user, I would like to be able to reset a forgotten password	Medium	Accounts	functional
BL-54	As a site maintainer, I would like multiple erroneous password attempts to result in a locked account for the user.	Medium	Security	non-functional
BL-55	As a site maintainer, I would like the user passwords to be stored in an encrypted fashion.	High	Security	non-functional
BL-56	As a site maintainer, I would like session IDs to be non-predictable and follow best practices per authoritative sources (e.g. OWASP, SANS)	High	Security	non-functional
BL-57	As a site maintainer, I want a dedicated administrative interface to perform admin-related tasks on the system.	High	Admin	non-functional

Table 1: Product Backlog

Sprint Backlog

The table below is the backlog for the first sprint. This sprint is focused on getting the main portions of application architecture off the ground, as specified in the architectural requirements, as well as some basic application functionality that will be required for the remainder of the application to function properly. This functionality is focused on the user account request process.

The requirements in Sprint 1 are deemed the highest priority of the entire backlog and will need to be implemented to allow further development to proceed. In other words, the Sprint 1 requirements are the critical path for the project and need to be addressed first.

Issue key	Summary	Priority	Area	Requirement Type	Sprint	Story point estimate	Status Category
BL-1	As a site developer, I would like the site to make use of a modern, actively-developed, secure framework	High	Architecture	non-functional	BL Sprint 1	10	In Progress

BL-2	As a site developer, I would like the site to use a database implementation that is modern, secure, actively-developed, and compatible with the selected web development framework	High	Architecture	non-functional	BL Sprint 1	5	To Do
BL-3	As a site maintainer, I need to update the vehicle make, model, and options as OEM offerings change	High	Architecture	non-functional	BL Sprint 1	10	To Do
BL-7	As a site user, I want to be able to set and change my password	High	Accounts	functional	BL Sprint 1	4	To Do
BL-8	As a site user, I want to sign up for an account	High	Accounts	functional	BL Sprint 1	7	To Do
BL-9	As a site user, I want to log into the application	High	Accounts	functional	BL Sprint 1	4	To Do
BL-10	As a site user, I want to log out of the application	High	Accounts	functional	BL Sprint 1	2	To Do
BL-11	As a site user, I want the ability to sign up for an account either as a car shopper or a car dealer.	High	Accounts	functional	BL Sprint 1	2	To Do

Table 2: Sprint 1 Backlog

Previously, an estimate for this sprint was calculated using the Use Case Points method. The total hours estimation from that exercise was calculated to be 1080 hours.

This week, the following method was used to attempt to estimate the time needed to implement the requirements in the sprint. First, each story in the sprint was assigned a number of story points. The number itself is not as important as the ratio of each story's story points relative to each other. For example, story A may be given 10 story points and story B 20. In this case, the important detail is that the story B is viewed as twice the effort as story A.

It is hoped that once the sprint is completed, the story point estimation will provide a useful metric on developer velocity for future sprint planning efforts. We will be able to see how many story points were completed in the sprint and use that as a guide for planning the next one. The sprint table above shows the story point estimate for each story in Sprint 1.

Once the story points are set for each story in the sprint, they are broken down into tasks. Since the story itself is written at the user level, it is often convenient to decompose the broad need of the user into smaller tasks that can be more easily consumed by the developer. In Jira, these are called Subtasks

and they are associated with the higher-level story. Each story can therefore have zero or more subtasks that define the work needed to complete that story in sufficient detail for a developer to work on it.

Once the subtasks are created, they can be estimated individually in developer time (hours). Each subtask gets assigned an “original estimate” in hours. Jira sums the estimate for each subtask into the main story they are associated with, so that a summation of the total estimate in time is viewable at the story level.

Once a developer starts work on a subtask within story, it gets moved in the sprint board from “To Do” to “In Progress”. As they work, the developer should keep track of the time spent on each subtask. They can periodically update the subtask time tracking field to indicate how much time has been spent on that task. The time worked data will be used to inform the accuracy of the subtask estimation exercise and lead to refinement in future sprints.

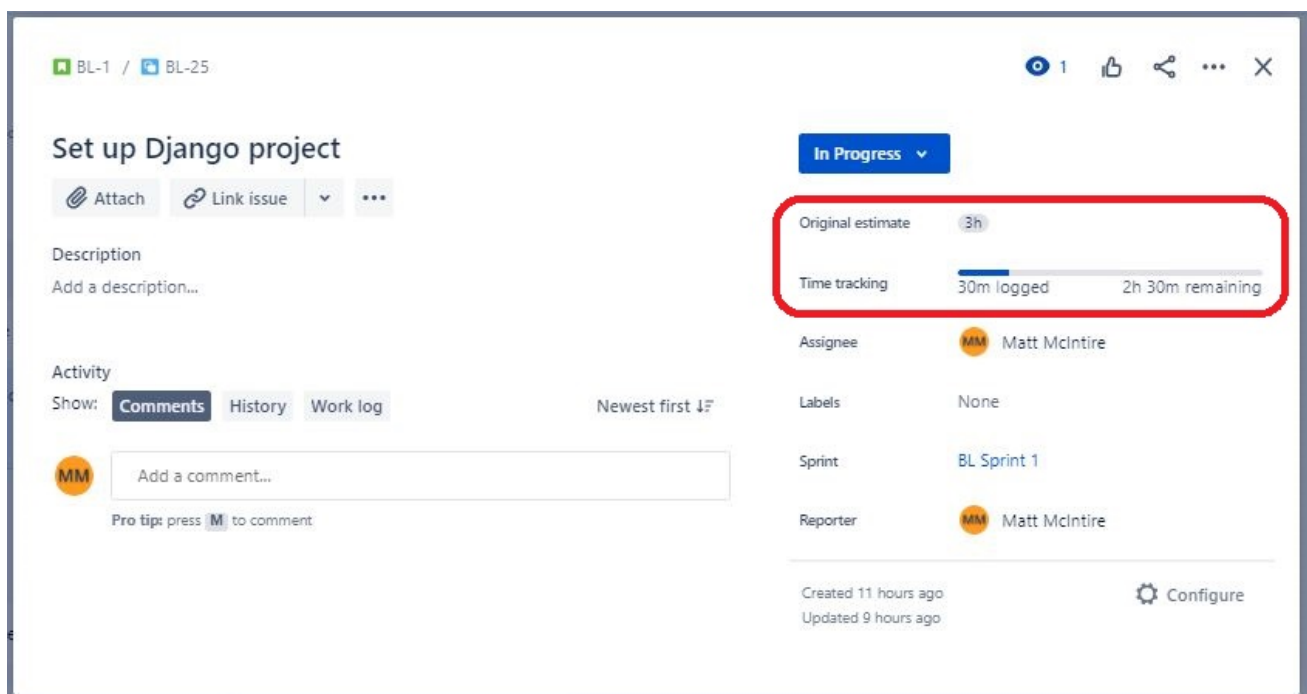


Figure 1: Jira subtask showing time estimation and tracking

As a site developer, I would like the site to make use of a modern, actively-developed, secure framework

Attach Add a child issue Link issue

Description

Django was chosen as the framework to meet this requirement. This story and its subtasks are written to provide the initial functionality needed to get the project off the ground.

Child issues

0% Done

- BL-25 Set up Django project IN PROGRESS
- BL-26 Stub out basic landing page TO DO
- BL-27 Get project checked into GitHub repo TO DO

In Progress

Time tracking 30m logged 1d 1h 30m remaining

Include child issues

Priority High

RequirementType non-functional

AppArea Architecture

Assignee MM Matt McIntire

Labels None

Sprint BL Sprint 1

Story point estimate 10

Figure 2: Jira story showing the time tracking roll-up and story point estimation

The following pages go through each story in Sprint 1, discuss its subtasks, and the effort estimation for each.

BL-1	As a site developer, I would like the site to make use of a modern, actively-developed, secure framework				
Description	Django was chosen as the framework to meet this requirement. This story and its subtasks are written to provide the initial functionality needed to get the project off the ground.				
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-25	Set up Django project	Subtask	Done	3h
	BL-26	Stub out basic landing page	Subtask	In Progress	5h
	BL-27	Get project checked into GitHub repo	Subtask	Done	2h
	BL-77	Testing – framework and repo test	Subtask	To Do	2h
Story Points	10		Sprint	BL Sprint 1	
Requirement Type	Non-Functional		Area	Architecture	

BL-3	As a site maintainer, I need to update the vehicle make, model, and options as OEM offerings change				
Description		Provide some sort of interface to allow maintainers to update the vehicle makes, models, and options as needed.			
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-72	Design data model for representing vehicles	Subtask	To Do	3h
	BL-73	Implement data model in Django models...	Subtask	To Do	3h
	BL-74	Expose data model to the admin site	Subtask	To Do	1h
	BL-75	Testing - data model updates via admi...	Subtask	To Do	1h
	BL-76	Testing - vehicle data model	Subtask	To Do	4h
Story Points	10		Sprint	BL Sprint 1	
Requirement Type	Non-Functional		Area	Architecture	

BL-7	As a site user, I want to be able to set and change my password				
Description		Create a means of allowing users to set password (as part of creating an account) and change it afterwards as desired.			
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-24	Testing - set and change passwords	Subtask	To Do	2h
	BL-63	design password change page	Subtask	To Do	2h
	BL-64	Implement password change backend logic	Subtask	To Do	3h
	BL-65	Implement password set backend logic	Subtask	To Do	3h
Story Points	4		Sprint	BL Sprint 1	
Requirement Type	Functional		Area	Accounts	

BL-8	As a site user, I want to sign up for an account				
Description	Create an account sign up page and the associated backend functionality.				
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-58	Create account signup template	Subtask	To Do	2h
	BL-59	Design account signup page	Subtask	In Progress	2h
	BL-60	Create account data model	Subtask	To Do	1h
	BL-61	Add account registration logic in bac...	Subtask	To Do	2h
	BL-62	Testing - account registration	Subtask	To Do	4h
Story Points	7		Sprint	BL Sprint 1	
Requirement Type	Functional		Area	Accounts	

BL-9	As a site user, I want to log into the application				
Description	Add UI elements and associated backend functionality to allow users to log into the application and get an active session.				
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-28	Create log in page template	Subtask	To Do	1h
	BL-29	Design login page	Subtask	To Do	2h
	BL-78	Testing – users can log into the system	Subtask	To Do	1h
Story Points	4		Sprint	BL Sprint 1	
Requirement Type	Functional		Area	Accounts	

BL-10	As a site user, I want to log out of the application				
Description	Add UI elements and associated backend functionality to allow users to log out of the application and have their session destroyed.				
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-66	Implement backend logic to destroy session and logout users	Subtask	To Do	2h
	BL-67	Testing - log out feature	Subtask	To Do	1h
	BL-68	Design and create logout UI elements ...	Subtask	To Do	2h
Story Points	2	Sprint	BL Sprint 1		
Requirement Type	Functional	Area	Accounts		

BL-11	As a site user, I want the ability to sign up for an account either as a car shopper or a car dealer.				
Description	Provide a means in the UI to allow users to select whether they are a car shopper or dealer.				
Subtasks	Key	Summary	Type	Status	Effort Estimation
	BL-69	Design and implement web UI controls to differentiate between a shopper and dealer	Subtask	To Do	2h
	BL-70	Implement backend logic to differentiate account types	Subtask	To Do	2h
	BL-71	Testing - account types (dealer vs shopper)	Subtask	To Do	2h
Story Points	2	Sprint	BL Sprint 1		
Requirement Type	Functional	Area	Accounts		

BL-2	As a site developer, I would like the site to use a database implementation that is modern, secure, actively-developed, and compatible with the selected web development framework		
Description	Provide a means in the UI to allow users to select whether they are a car shopper or dealer.		
Subtasks	N/A		
Story Points	5	Sprint	BL Sprint 1
Requirement Type	Non-Functional	Area	Architecture

Testing Plan

This section outlines the testing plan for the development of BottomLine. Each story in the active sprint will have one or more unit tests developed as part of the verification of the unit level code. The unit test is included as a subtask in each story, as shown in the previous section.

Below, each testing subtask is discussed. At this early stage of development, most tests for the current sprint have not been created. Each week, progress on these tests will be examined.

BL-77	Testing – framework and repo test		
Description	Develop tests as appropriate to ensure the framework and code repository are operating as intended. Check that the repository can be cloned into a new development environment and deployed successfully.		
Test type	Functional		
Preconditions	<ul style="list-style-type: none"> Stable release has been tagged or merged into master branch. System with required development tools needed to deploy the source has been identified. 		
Postconditions	<ul style="list-style-type: none"> Tester can navigate to BottomLine landing page for the local deployment completed. 		
Test Steps	TBD		
Status	To Do		

BL-75	Testing - data model updates via admin site
Description	Test that an admin user can update the data for a particular data model via the admin site.
Test type	Functional
Preconditions	<ul style="list-style-type: none"> At least one data model has been created and added to the application.
Postconditions	<ul style="list-style-type: none"> Tester can navigate to BottomLine admin page and see the data model under test is present on the page.
Test Steps	TBD
Status	To Do

BL-76	Testing – Vehicle data model
Description	Verify that a vehicle make, model, and option are valid. Ensures the dependencies and relationships between the tables are established correctly.
Test type	Unit
Preconditions	<ul style="list-style-type: none"> Vehicle data model information is loaded in the BottomLine database for the system under test.
Postconditions	<ul style="list-style-type: none"> Tester verifies the test passes. The test validates one or more known vehicle configurations (make, model, and option(s)) against the data model
Tests	TBD
Status	To Do

BL-24	Testing - set and change passwords
Description	Verify a user can successfully set and change their password.
Test type	Unit
Preconditions	<ul style="list-style-type: none"> Accounting subsystem is in place
Postconditions	<ul style="list-style-type: none"> Tester verifies the test passes. The test validates one or more test users can set and change a password
Tests	<ol style="list-style-type: none"> Verify password rules are enforced for any password submission Verify the password entries are confirmed to match when submitted For a password change, verify the new password is different from the previous password.
Status	To Do

BL-62	Testing – account registration
Description	Verify a user can sign up for a new account
Test type	Unit
Preconditions	<ul style="list-style-type: none"> Accounting subsystem is in place
Postconditions	<ul style="list-style-type: none"> Tester verifies the test passes. The test validates one or more test users can create a new account
Tests	<ol style="list-style-type: none"> Verify a new account can be created Verify an account creation is rejected if the user email is already in the system. Verify account request fields are validated
Status	To Do

BL-78	Testing – users can log into the system
Description	Verify a user can log into the system and get an active session
Test type	Unit
Preconditions	<ul style="list-style-type: none"> Accounting subsystem is in place
Postconditions	<ul style="list-style-type: none"> Tester verifies the test passes. The test validates one or more test users can log into the system
Tests	<ol style="list-style-type: none"> Verify a test user can log into the system Verify submission of a valid username and password result in a successful login. Verify submission of an invalid username results in a failed login Verify submission of an invalid password for a valid username results in a failed login.
Status	To Do

BL-67	Testing – log out feature
Description	Verify a user can log out of the system and that the associated session is destroyed
Test type	Unit
Preconditions	<ul style="list-style-type: none"> Accounting subsystem is in place
Postconditions	<ul style="list-style-type: none"> Tester verifies the test passes. The test validates one or more test users can log out of the system
Tests	<ol style="list-style-type: none"> Verify a test user can log out of the system Verify a logout action results in destruction of the session ID in BottomLine backend. Verify a logout action results in redirection to the landing page.
Status	To Do

BL-71	Testing – account types
Description	Verify a user can sign up for either a shopper or dealer account type.
Test type	Unit
Preconditions	<ul style="list-style-type: none"> Accounting subsystem is in place
Postconditions	<ul style="list-style-type: none"> Tester verifies the test passes. The test validates account types are supported
Tests	<ol style="list-style-type: none"> Verify account request links take users to the appropriate sign up page for their selected account type. Verify the account type is retained in the BottomLine backend.
Status	To Do

Burndown Chart

The following chart is taken from the Jira BottomLine project. It is available at the link below:

<https://team-1622159142308.atlassian.net/jira/software/projects/BL/boards/1/reports/burndown>

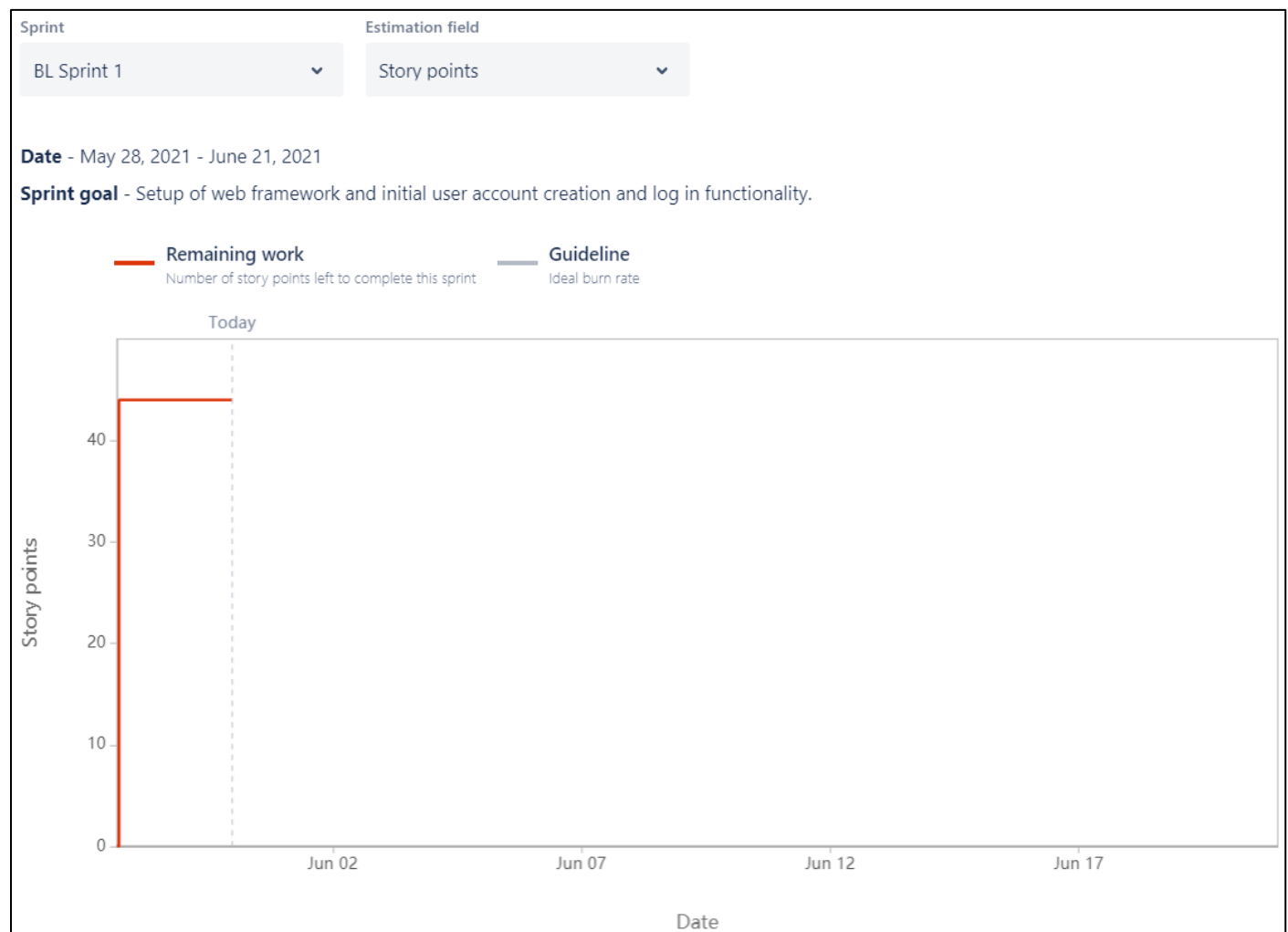


Figure 3: Sprint 1, week 1 burndown chart

Notes:

As shown in Figure 3, the burndown is flat through week 1. While some subtasks were completed, no stories were able to be closed out, resulting in a flat chart. Measurable progress against the established stories and subtasks was difficult this week. Specific challenges include:

- Developer time needed to learn git version control, Django, and PyCharm (IDE) for this project.
- Developer time needed to study best practices for getting the above-mentioned tools set up and working together.
- Time spent learning and setting up Jira to create a new project and enter current sprint and backlog stories into the tool.
- Time spent decomposing each Sprint 1 story into subtasks.
- Consideration for the testing plan for the project and generation of this document.

It is anticipated that development work will increase in the next weeks as the administrative burden of getting the project started diminishes and the developer moves further along the learning curve for the tools needed.

Key	Summary	Issue type	Epic	Status	Assignee	Story points
BL-1	As a site developer, I would like the site to make use of a mo...	Story		IN PROGRESS	MM	10
BL-2	As a site developer, I would like the site to use a database im...	Story		TO DO	MM	5
BL-3	As a site maintainer, I need to update the vehicle make, mod...	Story		TO DO	MM	10
BL-7	As a site user, I want to be able to set and change my passw...	Story		TO DO	MM	4
BL-8	As a site user, I want to sign up for an account	Story		IN PROGRESS	MM	7
BL-9	As a site user, I want to log into the application	Story		TO DO	MM	4
BL-10	As a site user, I want to log out of the application	Story		TO DO	MM	2
BL-11	As a site user, I want the ability to sign up for an account eit...	Story		TO DO	MM	2

Figure 4: Sprint 1, week 1 incomplete stories

Reference Materials

The following are links to referenced resources for the project.

- Source code repository: <https://github.com/mcm219/BottomLine>
- Jira project: <https://team-1622159142308.atlassian.net/jira/software/projects/BL/boards/1/backlog>