Planning of the testing activities

Assumptions made:

1. The legacy system is still actively used,the new system is only being built and not used by end users yet, which makes the working legacy system the first priority.
2. There is a lot less testing work for the legacy system because it's fully covered by the automation tests (I assume that the number of failing tests is not too large).
3. Features are being developed for the legacy system as well but they are a lot fewer as this system is legacy and the support will probably end rather soon in the future.
4. The support of the legacy system will end some time in the future but it's not going to happen in the nearest few weeks.
5. The legacy system and the new system are 2 different applications and the existing automation tests can't be used to test the new system.

Challenges:

1. Prioritization and time management, as there 2 applications that need to be supported at the same time.
2. Implementation of test automation while releases need to keep going smoothly.
3. New system is probably not that stable yet.

Plan of action:

Since this is the first week I'd start from taking a look at what is currently being done for the legacy and the new systems and evaluate the amount of work that need to be regularly performed there.
Based on that I'd start preparing a test plan describing the testing process.
At first I'd split my time approximately 30-40% for the legacy system and 60-70% for the new system, revaluate this proportion and the priorities each spring and correct if needed.

At the beginning of the week there are probably no tickets in testing yet. This time can be used for test automation.
Then I'd start with the regression tests for the legacy system. First of all the analysis of the failing tests needs to be performed and depending on the reason of the failure either bugs need to be reported or the automation tests need to be updated to have reliable regression runs further on.

When new features for the legacy system go to testing, first they should be manually tested. The defects need to be fixed and validated. After that when the feature is accepted and stable it can be automated. The point when we consider the feature ready for automation can be agreed with the team.
Possible option is to create a test automation task for this feature which will be picked up next sprint (after it has been released).

For the new system the normal way of working would be quite similar. New features would first be verified manually. And when they are accepted, they will proceed to automation.

However, since it's my first week and there is no automation for the new system, I'd start with investigation of the requirements, tech stack and architecture. Based on that I'd define the set of tools that we are going to use for the test automation.
Next step would be to do a basic set up of the test automation project and implement a few tests as a proof of concept and present it to the team.
The solution will be discussed within the team, changes will be applied if needed. When we reach an agreement within the team on the approach we can continue developing the test automation project increasing the coverage and improving the set up in the following sprints.

Summary:

The priorities would be set in the following manner between the applications:
1. Legacy system
2. New system

Among the types of work:
1. Testing of new features and defect validation
2. Support of the existing automation tests (analysis and fixes)
3. Selection of tools and set up of the test automation project for the new system
4. Implementation of new scenarios for the legacy system
5. Implementation of new scenarios for the new system
6. Framework and infrastructure improvements

The above mentioned plan, would be described in detail in a test plan, agreed with the team and published on the company's Wiki.