

CSC-580 PM CASE: THIRD AVENUE SOFTWARE HEALTH-CARE APP PROJECT

This case is new for the ninth edition of *Information Technology Project Management*. The case provides an opportunity to apply agile and Scrum principles to project management.

Each part of the case contains several task assignments to help you explore the use of agile and Scrum principles.

Part 4: Project Cost and Quality Management

As with other methods of software development, the application's size is a major indicator of how much it might cost to develop. In agile development, cost estimates are often made based on size measurements such as story points.

You will recall that each user story for the health-care app is assigned several story points based on estimates of how much work is needed to complete each major task. Based on prior experience with agile projects, the three-person accounting staff at Third Avenue Software has determined an average dollar production cost for a story point: \$1200.

However, the staff accountants are not completely confident in this average dollar value because Third Avenue's experience with agile projects is not extensive. The accountants would like the team to confirm their calculations, if possible.

As you learned in Module 7, earned value management (EVM) is a more traditional project management method for determining whether a project is meeting time and cost goals. EVM requires calculation of three values for each major activity in the project:

- The planned value (PV), which is the authorized budget assigned to scheduled work.
- The actual cost (AC), which is the realized cost for the work performed on an activity during a specified timeframe.
- The earned value (EV), a measurement of project work done, framed in terms of the approved budget for the work.

Several more values and formulas are involved in EVM. For more information, refer to the "Earned Value Management" section of the Module 7 Reading.

Managers at Third Avenue are also eager to see evidence that the Quality Assurance staff are making progress in their ability to test the health-care app. They have asked the team to provide at least a basic framework of test specifications.

- *Open a new Microsoft® Word document and complete the **Tasks** below.*
- *Save the file on your computer with your last name in the file name. (Example: part 1 tasks _Jones.doc)*
- *Click the **Choose File** button to select and upload your saved document.*

Tasks

1. Review the user stories, technical stories, and story points you developed in earlier parts of this case and then use the information to estimate the cost of the health-care app project. As you learned in the introduction to Part 4, Third Avenue accountants estimate that the average production cost of a story point is \$1200. Based on the values you calculate, do the project costs appear to be within the overall budget of \$350,000? Do the costs seem to be within the MVP budget of \$120,000? If the costs are not within budget, adjust the story points you developed in Part 3 to make sure your revised numbers are within budget. You might also have to reconsider the feature set within the MVP if it exceeds its budget.

2. Assume that you have completed one month of the health-care app project and have some reliable EVM data for cost accounting. Remember that the budget at completion (BAC) is \$350,000 for the four-month project. You have received the following figures from the Third Avenue accountants:

PV \$105,000

EV \$122,000

AC \$105,000

Using this information, answer the following questions.

- What is the cost variance, schedule variance, cost performance index (CPI), and schedule performance index (SPI) for the project?
 - Use the CPI to determine the estimate at completion (EAC) for the project.
 - Based on your answers, does the project appear to be on schedule and within budget?
3. The health-care app will undergo rigorous software testing by the Third Avenue Quality Assurance staff, using test plans that might fill entire binders. At this early point in the process, however, such plans are still in development. Develop a short list of quality requirements for testing at least five of the important app features and/or usability issues described thus far in this running case. In your list, briefly describe each requirement.

For example, here is a short (and far from complete) list of quality requirements for the emergencies list feature, which stores vital phone numbers and addresses as well as provides quick access to local in-network hospitals, urgent care clinics, and children or friends who can be relied upon to provide transportation in an emergency:

- Do all the data-entry fields accept text without problems? For example, long entries in a field can sometimes be clipped or truncated prematurely, which is frustrating to users.
- Is it possible to enter nonsensical data in a field? For example, users should not be allowed to type text characters into a field that records phone numbers.

- Does the software automatically move the user from one entry field to the next? Does the cursor always appear in the next entry field as a means of visual feedback for the user?
 - As soon as the user enters and saves an address, an interactive GPS map should become available in a new window, with voice and text directions. Does the window appear? Does it appear immediately?
 - When you slide your finger to create the effect of movement within the map, does the software redraw in a timely manner?
4. Modules 2 and 8 of the text discussed Kanban, a method sometimes used with Scrum. Kanban uses five core properties:
- *Visualize the workflow.* Cards are a common visualization method.
 - *Limit work in progress.* New project work is done when it can be incorporated into the next software iteration and when there is available capacity. In this way, problem areas are quickly revealed for resolution.
 - *Measure and manage flow.* A key here is to analyze problem areas and then implement changes to correct the problems as quickly as possible.
 - *Make process policies explicit.* Everyone on the project team must understand the processes and any problems with them.
 - *Use models to recognize improvement opportunities.* The models themselves are less important than the important agile concept of continuous improvement.

Based on your work on the case so far, which property of Kanban has proven most useful to you? Write a two- to three-paragraph answer.