

## PROJECT 3.1.9

# Social Networking App - Design

### INTRODUCTION

In the previous activities, you created the Trip Tracker app with a cloud-based back end, user authentication features, and trip management functionality. You also used different tools to help you design solutions and communicate them with others. In this project, you will use proto.io to create a mockup of a simple social networking app, which you will build later in this lesson.

#### Materials

- Computer with Internet connection
- Free proto.io account per student

### RESOURCES



**Lesson 3.1 Reference Card for Backendless**  
Resources available online

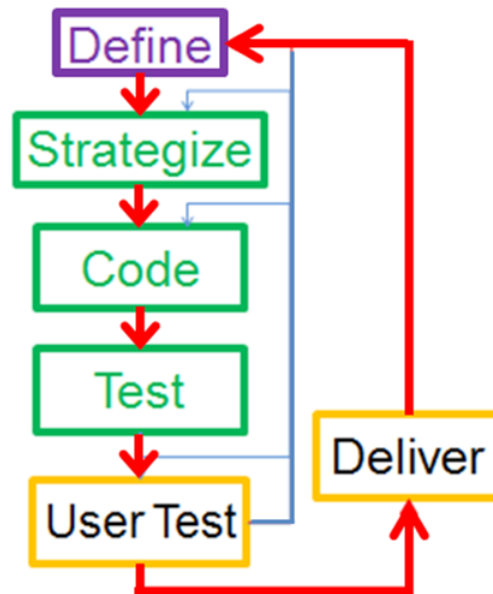
## Procedure

- 1 Form pairs as directed by your teacher. Recall that in pair programming, one person is the driver—the one using the computer, coding, and explaining his/her reasoning, and the other person is the navigator—the one asking questions, suggesting alternatives, and catching any typos. While you work on this project, make sure to switch roles as directed by your instructor.

- 2 Refer to the Agile software development process described in the following slideshow. In this project you will work on the Define and Strategize steps.

## Software Development Process

1. Define
2. Strategize
3. Code
4. Test
5. User Test
6. Deliver



Computer Science A

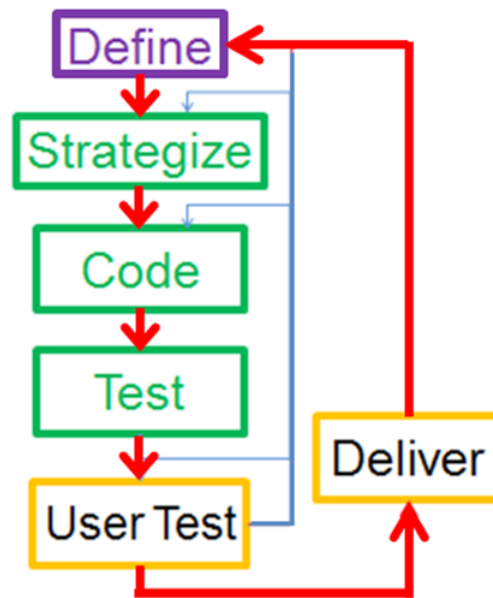
© 2016 Project Lead The Way, Inc.

Most software development processes involve similar steps:

1. Define the problem to be solved. This includes asking questions, interviewing people, and understanding the requirements.
2. Strategize includes exploring options, brainstorming ideas, and designing potential solutions.
3. Code the solution.
4. Test the solution to make sure it solves the problem identified in the first step.
5. Users test the solution to make sure it works as expected. This is also known as Quality Assurance (QA) testing.
6. Deliver the solution.

# Agile Methodology

- Frequent deliverable iteration
- One cycle called a **sprint**



Computer Science A

© 2016 Project Lead The Way, Inc.

In Waterfall software development, the steps are linear. One step has to be completed before the next one begins. This method usually takes longer to deliver a solution and does not involve the client after the problem is defined in step 1.

In Agile software development, the process is not linear. Instead it generates frequent deliverables as part of an iterative process. Once a problem is defined, programmers go through frequent cycles of developing the solution in increments and presenting the results to the client. This way, the client can help clarify the specifications of the software as it is being built. This also allows for faster delivery. Agile is commonly used in Industry today.

Each cycle in Agile is called a sprint (or iteration) and may last one to four weeks for a professional team. Programmers code and test multiple times and even may have to re-strategize in the process, then code and test. The cycle repeats until they are ready to share with the client for feedback. Then they start another sprint, in which they work on another set of requirements or refine some from their previous sprint.

# Organizing Agile Work

## Backlog

- ☐ User Story #1
- ☐ User Story #2
- ☐ User Story #3
- ☐ User Story #4
- ☐ Etc...

- Prioritized
- Top of backlog well defined

Computer Science A

© 2016 Project Lead The Way, Inc.

Programmers usually organize their work using two lists: backlog and task list.

The backlog is the list of requirements identified by the client and organized from highest to lowest priority. The items in this list are usually written in an informal fashion and are referred to as user stories. User stories should always describe the way the user interacts with the software application. For example, “the user should register to use the app”. During each sprint, programmers work on one or two user stories, based on the time they estimated it would take to develop each one (see next slides).

## Backlog Example

### Backlog

- ☐ Students should register with a school to use the app.
- ☐ Students can enroll in multiple classes.
- ☐ Teachers should be able to post assignments for each class.
- ☐ Students should be able to submit their assignments back to the teacher.
- ☐ Students should be able to view their grades for each class.

Computer Science A

© 2016 Project Lead The Way, Inc.

The example shows a backlog for a project that aims to develop an app to support a student learning management system. The list consists of user stories that describe the student and teacher experiences.

# Organizing Agile Work (cont.)

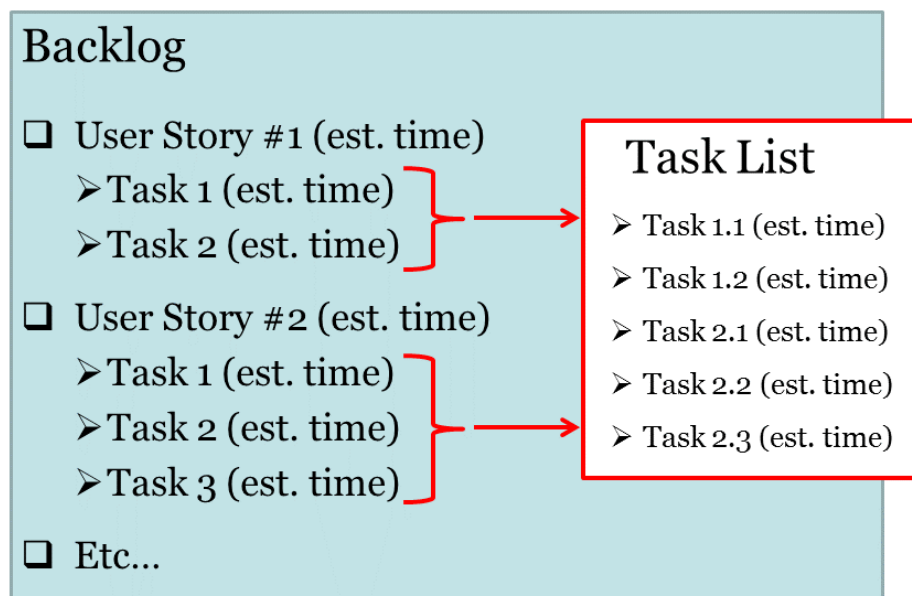
Each sprint or iteration includes:

- Iteration Planning
- Iteration Execution
- Iteration Retrospective

Computer Science A

© 2016 Project Lead The Way, Inc.

## Sprint step 1: Iteration Planning



Computer Science A

© 2016 Project Lead The Way, Inc.

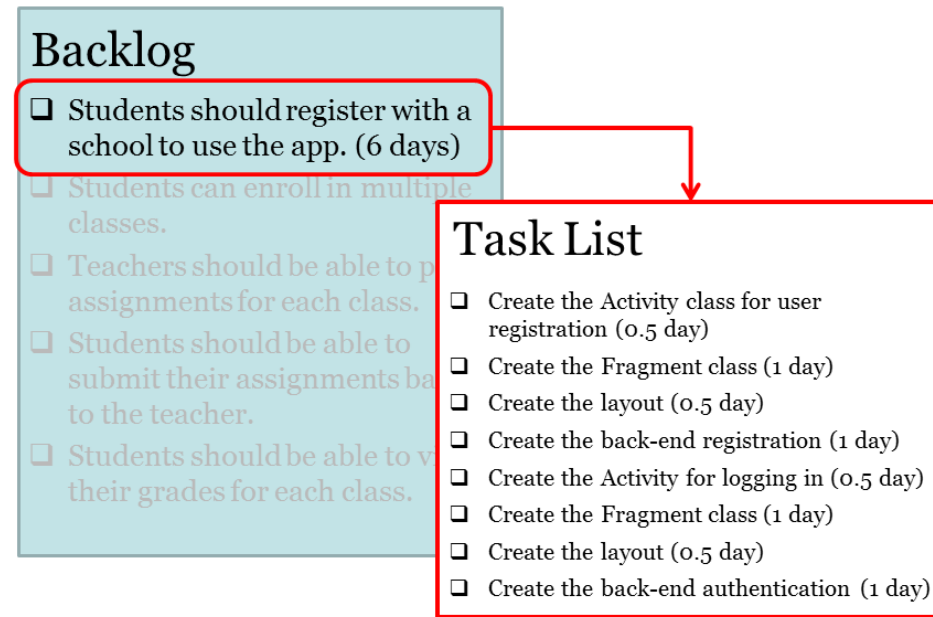
The first step of a sprint is the iteration planning step. Programmers begin to plan their work for the first sprint as follows:

1. Estimated the time to complete each user story. This will help identify how many user stories can be completed in an iteration.
2. Choose the user story (or stories) that they would like to complete in the first iteration (based on the estimated time.)
3. Translate each of the chosen user stories into a list of more specific tasks to be worked on. Tasks can be more technical and describe the nuts and bolts of what must be coded to fulfill the user story.
4. Estimate the time to complete each task. Tasks should be small enough to complete in a day or two. Any task that is too large should be broken into smaller tasks and re-estimated.

If a user story has a very large number of tasks, the tasks can be split up to work on the user story over two iterations.

The final list of tasks to be worked on during a sprint is called a task list.

# Task List Example



Computer Science A

© 2016 Project Lead The Way, Inc.

The task list above is an example of what tasks can be generated from the first user story. Note the time estimates on both the backlog and task list.

## Sprint step 2: Iteration Execution

- Divide task list among programmers.
- Start developing:
  - Code
  - Test
  - Do code review
  - Create Javadoc
- Present to client for feedback at the end of the sprint.

Computer Science A

© 2016 Project Lead The Way, Inc.

## Sprint step 3: Iteration Retrospective

- What did we do well?
- What did we NOT do well, and how should we address that in the future?

Computer Science A

© 2016 Project Lead The Way, Inc.

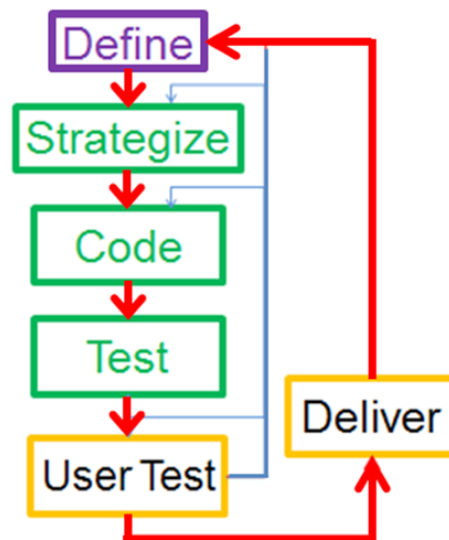
The retrospective is an important part of the whole sprint. The retrospective reflects on two main questions:

1. What did we do well? Identify successes that can be repeated in the future.
2. What did we NOT do well and how should we address them? Identify failures and lessons learned.

The retrospective enforces the core benefit of iterative development. Bringing what is identified during a retrospective to the next iteration planning is the “magic” that makes Agile so effective.

## Organizing Agile Work (cont.)


- Groom the backlog.
- Generate new task list.
- Repeat sprints until product is complete.



Computer Science A

© 2016 Project Lead The Way, Inc.

Last, the backlog is groomed (revisited and updated), and a new task list is generated to clearly define the tasks to be worked on in the next sprint. The cycle of sprints continues until the complete product is delivered.

- 3 Refer to the  **3.1.10 Requirements** document for a list of requirements for your product.
- 4 Work with your partner to brainstorm a list of features that you would like to incorporate into your social networking app. Follow the guidelines for brainstorming: go for quantity; never criticize ideas; “piling on” is welcome.

- 5 Create your backlog and prioritize it with the user stories that are most important (“must-haves”) at the top of the list and those that are optional (“nice-to-haves”) at the bottom of the list. Use a checklist similar to this format:

Backlog
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

- 6 Iteration Planning:
- (Optional) Add time estimates to each user story.
  - Break each of the high priority user stories into a number of smaller technical tasks.
  - Add a time estimate for each task.
  - Create a task list from the tasks identified. Your first sprint will focus on these tasks.

User Story Sprint Task List
1.
2.
3.
4.

- 7 Log in to proto.io and create a new project named **SocialNetworking**. If you are using the “Free Forever” version, you may need to delete your previous project. For prototyping, use the Android device **S. Galaxy S5**.
- 8 Develop a prototype for your first sprint.
- As you develop the prototype, keep in mind the following:
    - What screens are needed?
    - What components should go on those screens?
    - How will a user navigate from one screen to another?
    - How does the design affect the usability of the app?
  - Remember that one of the objectives of designing a solution is to flush out any further questions and/or requirements that surface as you work on your design. Therefore, make sure to update your backlog and/or task list as you work on your prototype.



- 9 Consider security issues as they apply to a Social Networking app.
  - a. Social Networking is a convenient and widely popular communication tool. But information posted on these sites can be abused by unethical users. **Social engineering** is the term used when criminals try to learn your personal information, usually found on social networking sites, and use it to target attacks specifically to you. You may receive unwanted emails, see popup messages asking you to follow links to malware sites, or be invited to install software you did not request. Protect yourself by dismissing or deleting suspicious emails, popups, ads, and messages. You can also protect yourself by typing URLs directly into a web browser, being careful to avoid typos.
  - b. Describe, as a sprint task, how to build a feature into your app that would help protect against social engineering abuse. (You will not be required to implement this sprint task.)
- 10 Present your prototype as directed by your instructor.

## CONCLUSION

1. Did the backlog and task list help or hinder your app design? Explain your thoughts.
2. Describe how the development of your prototype helped evolve your app.