Presented by:

**Team Caliware Hang, Joanna**

Timothy Cioffi-Dinkel

Anthony Farina

Joanna Hang

Lourdes Lopez

David Sullivan

**Managing a simulated agile project**

**CPSC 546-50 Instructor: Dr. Bin Cong**

May 13, 2016

# Table of Contents

[Table of Contents 2](#_Toc450927005)

[Envision Phase 4](#_Toc450927006)

[Vision Statement 4](#_Toc450927007)

[Project Objectives and Constraints 4](#_Toc450927008)

[Speculate Phase 7](#_Toc450927009)

[User Stories 7](#_Toc450927010)

[Release Plan 7](#_Toc450927011)

[Initial Product Backlog 7](#_Toc450927012)

[Initial Release Plan 9](#_Toc450927013)

[Timeline 11](#_Toc450927014)

[Explore Phase 12](#_Toc450927015)

[Sprint Planning Meeting 12](#_Toc450927016)

[Sprint Backlog 12](#_Toc450927017)

[Task Board 12](#_Toc450927018)

[Acceptance Criteria 12](#_Toc450927019)

[Continuous Integration 12](#_Toc450927020)

[Burn Down Chart 12](#_Toc450927021)

[Daily Scrum Meeting 13](#_Toc450927022)

[Release Burn Down Chart 13](#_Toc450927023)

[Sprint Review Meeting 13](#_Toc450927024)

[Adapt Phase 14](#_Toc450927025)

[Sprint Retrospective 14](#_Toc450927026)

[Sprint One 15](#_Toc450927027)

[Sprint Planning Meeting 15](#_Toc450927028)

[Issue Tracking 17](#_Toc450927029)

[Sprint Burn Down Chart 18](#_Toc450927030)

[Retrospective (Adapt Phase) 18](#_Toc450927031)

[Sprint Review Meeting 19](#_Toc450927032)

[Sprint Two 21](#_Toc450927033)

[Sprint Planning (Adapt Phase) 21](#_Toc450927034)

[Issue Tracking 23](#_Toc450927035)

[Sprint Burn Down Chart 25](#_Toc450927036)

[Retrospective (Adapt Phase) 25](#_Toc450927037)

[Sprint Review Meeting 25](#_Toc450927038)

[Sprint Three 27](#_Toc450927039)

[Sprint Planning (Adapt Phase) 27](#_Toc450927040)

[Issue Tracking 29](#_Toc450927041)

[Sprint Burn Down Chart 31](#_Toc450927042)

[Retrospective (Adapt Phase) 31](#_Toc450927043)

[Sprint Review Meeting 32](#_Toc450927044)

[Final Product Backlog 34](#_Toc450927045)

[Final Release Plan 37](#_Toc450927046)

[Release Burndown Chart 37](#_Toc450927047)

[Velocity 37](#_Toc450927048)

[Close Phase 40](#_Toc450927049)

[Project Retrospective 40](#_Toc450927050)

[Our Process vs CMMI Project Monitoring Control and Project Planning 41](#_Toc450927051)

[Project Monitoring and Control 41](#_Toc450927052)

[Project Planning 42](#_Toc450927053)

[References 46](#_Toc450927054)

# Envision Phase

In the Envision Phase, the purpose is to establish the vision and set expectations for developing the product. The vision is crucial because all team members must understand the vision in order to understand the product that needs to be built and be aware of the shared goals of the stakeholders that we identified for this project. The activities in our Envision Phase include: create a Vision Statement and defining Project Objectives and Constraints.

The Vision Statement is defined by the elevator test statement. It is a short explanation of the project using the following format:

* For (target customer)
* Who (statement of the need or opportunity)
* The (product name) is a (product category)
* That (key benefit, compelling reason to buy)
* Unlike (primary competitor product)
* Our product (statement of primary differentiation)

In order to develop the initial baseline, the important project objectives and constraints are captured in the Product Data Sheet. This is a summary of the key elements: Clients/customers, Product manager, executive sponsor, and the tradeoff matrix that include the anticipated scope, schedule, and cost information.

## Vision Statement

For universities, who want to manage campus enrollment the Student Information System will maintain user roles, enroll students, and record grades. Unlike Banner by Ellucian, our product features include a Learning Management System, and cloud document storage, and graduation checks when students are nearing graduation.

## Project Objectives and Constraints

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Data Sheet** | | | | | | |
| **Project Name: Student Information System** | | | | | Executive Sponsor: Mildred Garcia | |
| Project Start Date: 05/03/2016 | | | | | Project Manager: Anthony Farina | |
| **Clients:** | | | | | Product Manager: David Sullivan | |
| California State University, Fullerton | | | | | Project Team: Joanna Hang, Lourdes Lopez, Timothy Cioffi-Dinkel | |
| CSUF Students | | | | | Government: State of California | |
| CSUF Faculty | | | | |  | |
| CSUF Staff | | | | | **Quality Objectives:** | |
| Admission and Records | | | | | Performance: Response time shall be less than 1 second | |
| Division of Academic Affairs | | | | | Reliability: No more than 5 minutes of downtime or failure | |
| The CSU | | | | | Security: faculty and staff password reset campaign every 4 months | |
|  | | | | | Availability: must be available 95% every day | |
| **Project Objective Statement:** | | | | |  | |
| The objective is to build a student information system that maintains user roles, enrolls students, and records grades. The system needs to be operational by 8/1/2016 and cost less than $500,000. | | | | |  | |
| **Performance Guidelines:** | |
| Volume of 40,000 student class registrations two months prior to semester start | |
|  | | | | | Volume of 40,000 student accounts and 3,000 faculty/staff accounts | |
| **Business Objectives:** | | | | | Volume of 11,000 classes | |
| Integrated Financial Aid Module will improve workflow between registrar and FA depts. | | | | | Internet access | |
| Email component will reduce processing time for administrative tasks | | | | |  | |
| Cloud storage will centralize student documents for use between various departments. | | | | |  | |
| Integrated LMS Module will reduce grading process for faculty. | | | | | **Architecture Guidelines:** | |
|  | | | | | Integrate with LMS | |
| **Trade-Off Matrix:** | | | | | Integrate with Cloud Storage | |
|  | Fixed | Flexible | Accept | Target | Integrate with E-mail Service | |
| Scope |  | x |  | 12,000 FP | Integrate with PeopleSoft (HR System) | |
| Schedule | x |  |  | 8/1/2016 |  | |
| Cost |  |  | x | "+/-" $100,000 |  | |
| Project Delay Cost per Month: $100,000 | | | | | **Major Project Milestones:** | |
| Exploration Factor: 9 | | | | | Accounts Management | 5/11/2016 |
|  | | | | | Class Registration | 5/23/2016 |
| **Capability:** | | | | | Transcript | 6/7/2016 |
| **Accounts management** | | | | |  | |
| Separate user roles for students, faculty, and staff | | | | | **Issues and Risks:** | |
| Assign passwords | | | | | Faculty reluctant to new student information system | |
| Authorize/unauthorize student and faculty accounts | | | | | Ability to meet deadline | |
| Enter class schedule and prerequisites | | | | | Resources and availability to develop system | |
| **Class registration** | | | | | May exceed budget due to complexity of system | |
| Add, drop classes | | | | |  | |
| Display prerequisites | | | | |  | |
| Show course offerings | | | | |  | |
| Show open sections | | | | |  | |
| **Transcript** | | | | |  | |
| Record grades | | | | |  | |
| See history of classes | | | | |  | |
| Student grade level | | | | |  | |
| Graduation checks | | | | |  | |

# Speculate Phase

The speculate phase involves creation of the Product Backlog and Release Planning. The customers were involved by adding User Stories which were facilitated by the Scrum Master and the Product Owner. This phase is not wholly focused on the software process. Rather, we insure that the customer requirements are discussed at a high level, but with well detailed User Stories. The User Stories were added to the Product Backlog and the priority was selected by the Scrum Team. These stories help the team extract requirements. User Stories were be collected during the speculative phase as the beginning of Scrum. Additionally, User Stories can be added at the beginning of each iteration. The Product Owner has the final authority for determining the priority and it is based on the stories that have the highest value.

### User Stories

User Stories are short and simple statements from the user that describe the functionality/feature that they expect the system or product to perform. User Stories are written in a template:

As a <who>, I want <what (feature)>, so that <why (value)>

From this user story, the team can talk to the user to get a more refined story and then proceed with creating acceptance tests for the user story.

### Release Plan

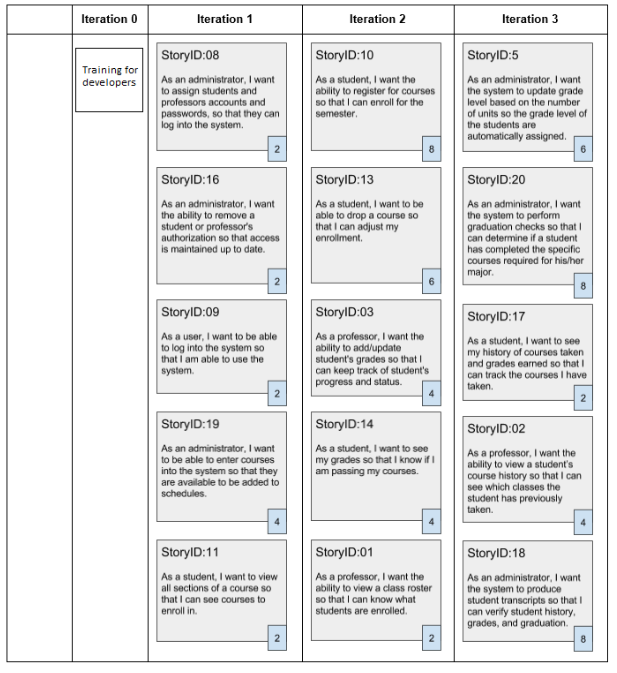
Developing a Release Plan is a collaborative team effort. From the items in the Product Backlog, the developers provided story points to provide estimates the work of each story. The developers used the story points to determine the work that can be done in each release. Then, the Scrum Team will divide the work based on the value and risk.

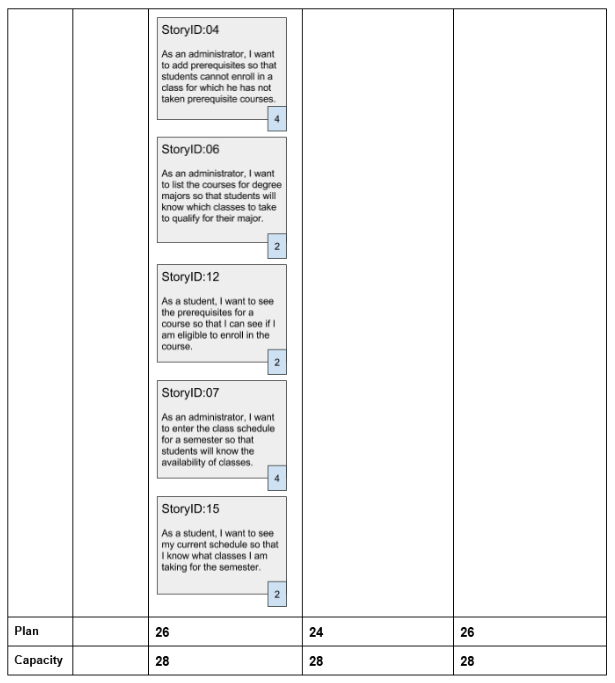
### Initial Product Backlog

The Initial Product Backlog was created during the speculate phase and was later reviewed prior to the beginning of each sprint.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Story Id** | **Story Name** | **Story** | **Priority** | **Planned Iteration** | **Estimation (S,M, L, XL)** | **Estimation Points** | **Dependencies** | **Acceptance Test** |
| 8 | Assign Accounts | As an administrator, I want to assign students and professors accounts and passwords so that they can log into the system. | 1 | 1 | S | 2 |  |  | |
| 16 | Remove account | As an administrator, I want the ability to remove a student or professor's authorization so that access is maintained up to date. | 2 | 1 | S | 2 |  |  | |
| 9 | Login | As a user, I want to be able to log into the system so that I am able to use the system. | 3 | 1 | S | 2 |  |  | |
| 19 | Add courses | As an administrator, I want to be able to enter courses into the system so that they are available to be added to schedules. | 4 | 1 | M | 4 |  |  | |
| 11 | View course | As a student, I want to view all sections of a course so that I can see courses to enroll in. | 5 | 1 | S | 2 |  |  | |
| 4 | Prerequisites | As an administrator, I want to add prerequisites so that students cannot enroll in a class for which he has not taken prerequisite courses. | 6 | 1 | M | 4 |  |  | |
| 6 | Degree majors | As an administrator, I want to list the courses for degree majors so that students will know which classes to take to qualify for their major. | 7 | 1 | S | 2 |  |  | |
| 12 | View course prerequisite | As a student, I want to see the prerequisites for a course so that I can see if I am eligible to enroll in the course. | 8 | 1 | S | 2 |  |  | |
| 7 | Add Schedule | As an administrator, I want to enter the class schedule for a semester so that students will know the availability of classes. | 9 | 1 | M | 4 |  |  | |
| 15 | View current schedule | As a student, I want to see my current schedule so that I know what classes I am taking for the semester. | 10 | 1 | S | 2 |  |  | |
| 10 | Register for Courses | As a student, I want the ability to register for courses so that I can enroll for the semester. | 11 | 2 | XL | 8 |  |  | |
| 13 | Drop Course | As a student, I want to be able to drop a course so that I can adjust my enrollment. | 12 | 2 | L | 6 |  |  | |
| 3 | Record Grades | As a professor, I want the ability to add/update student's grades so that I can keep track of student's progress and status. | 13 | 2 | M | 4 |  |  | |
| 14 | View grade | As a student, I want to see my grades so that I know if I am passing my courses. | 14 | 2 | M | 4 |  |  | |
| 1 | Class Roll | As a professor, I want the ability to view a class roster so that I can know what students are enrolled. | 15 | 2 | S | 2 |  |  | |
| 5 | Update Grade Level | As an administrator, I want the system to update grade level based on the number of units so the grade level of the students are automatically assigned. | 16 | 3 | L | 6 |  |  | |
| 20 | Graduation Check | As an administrator, I want the system to perform graduation checks so that I can determine if a student has completed the specific courses required for his/her major. | 17 | 3 | XL | 8 |  |  | |
| 17 | History | As a student, I want to see my history of courses taken and grades earned so that I can track the courses I have taken. | 18 | 3 | S | 2 |  |  | |
| 2 | Student History | As a professor, I want the ability to view a student’s course history so that I can see which classes the student has previously taken. | 19 | 3 | S | 2 |  |  | |
| 18 | Student Transcript | As an administrator, I want the system to produce student transcripts so that I can verify student history, grades, and graduation. | 20 | 3 | XL | 8 |  |  | |

### Initial Release Plan





### Timeline

|  |  |  |
| --- | --- | --- |
| **BEGIN** | **END** | **DESCRIPTION** |
| 2/21 | 2/27 | Planning Phase  Create initial product backlog |
| 2/28 | 3/5 | Planning Phase continued |
| 3/6 | 3/12 | Iteration 1  Iteration Planning |
| 3/13 | 3/19 | Iteration 1 |
| 3/20 | 3/26 | Iteration 1  Release 1  Sprint Review  Backlog Review  Sprint Retrospective  Refine Release Plan |
| 3/27 | 4/2 | Iteration 2  Iteration Planning |
| 4/3 | 4/9 | Iteration 2 |
| 4/10 | 4/16 | Iteration 2  Release 2  Sprint Review  Backlog Review  Sprint Retrospective  Refine Release Plan |
| 4/17 | 4/23 | Iteration 3  Iteration Planning |
| 4/24 | 4/30 | Iteration 3 |
| 5/1 | 5/7 | Iteration 3  Release 3  Sprint Review  Backlog Review  Sprint Retrospective  Refine Release Plan |
| 5/8 | 5/14 | Project Summary Report |

# Explore Phase

In Agile development, we place emphasis on responding to change over following a plan (Beck, et.al, 2001). In the absence of a “fixed” plan, we follow agile practices to maintain good coordination and communication to be able to develop the software in smaller sprints instead of one large sequential start-to-end process where any derivation of the plan is discouraged. Agile Scrum practices were utilized for planning the iterations while XP technical practices were used for the design, development and testing.

### Sprint Planning Meeting

Before each iteration, the Sprint Planning Meeting is completed. The Product Owner picks the User Stories that need to be implemented by refining and reprioritizing the Product Backlog and Release Backlog with the Scrum Team. These selections were added to the Sprint Backlog. The Product Owner together with the team members wrote a brief Sprint Goal of what the team plans to accomplish in the upcoming Sprint.

### Sprint Backlog

In order to estimate the User Stories within the Sprint, the developers broke down the User Stories into short, estimated development tasks. The development team members then discussed the time they need to work on the items selected from the Sprint Backlog. After which, the development team volunteers for the tasks. These tasks are added to the Task Board.

### Task Board

A Task Board can be in the form of a whiteboard, a section of a wall or shared in a spreadsheet. The Task Board consists of four columns, “Story”, “To Do”, “In Progress”, and “Done”. The rows consist of the user story and the tasks that need to be complete the User Stories for the iteration. Each column also includes the complexity of the task, hours estimated, hours remaining, and the assigned developer. The Task Board was filled in at the beginning of each iteration, and the tasks were moved from left to right as when the tasks were currently being worked on or completed.

### Acceptance Criteria

Acceptance Criteria are directly attached to a user story. These represent a certain outcome that the Product Owner expects. If the system has the desired output, then the user story will be complete. Unit tests were created by the development team as well as beta testing by the customer during the Sprint Review.

### Continuous Integration

After code was checked into the code repository, the code was frequently built and tested together by automated tests in order to find any integration errors as quickly as possible.

### Burn Down Chart

A Sprint Burn Down Chart is a visual summary of the days in the Sprint compared to remaining tasks that are stated in the Sprint Backlog.

* X axis = time (days)
* Y axis= outstanding work

The Sprint Backlog graph was updated throughout the iteration so the team could view the progress to the goal.

### Daily Scrum Meeting

During the one week Sprint, the development occurs. The Scrum Master held a Daily Scrum Meeting only with the Development Team. This is a 15-minute stand-up meeting for the development team to discuss the progress of the work in the Sprint Backlog. The team discussed the work that was completed, the work that will be completed, and if there are any impediments or blocks to development. During the Scrum, each member answered these questions:

* What have you done since the last Scrum?
* What will you do between now and the next Scrum?
* What is getting in the way (blocks) the iteration goals?
* Is there any tasks relevant to the current iteration that should be changed or needs to be added?

Any further discussions that require more time is done after the Scrum meeting to go into more detail. There is not enough time in the 15 minute meeting to discuss solutions in detail.

### Release Burn Down Chart

A Release Burn Down Chart is a visual summary of the Sprint compared to remaining tasks that are stated in the Product Backlog.

* X axis = Sprints
* Y axis= story points

This collection of metrics allows the team to see trends and adjust accordingly. The Sprint

Backlog graph was set up in the Sprint Planning stage and the beginning of each iteration/sprint. It was updated throughout the iteration so the team can view the progress to the goal.

### Sprint Review Meeting

At the end of each Sprint, there a Sprint Review Meeting for the Scrum Team in which there was a demo of the product to all the stakeholders. During this Sprint Review, the developers listen to feedback only, and no further commitments made. Instead, commitments were discussed at the next Sprint Planning Meeting.

# Adapt Phase

The evolutionary model of Agile relies on continuous feedback to have the ability to introduce changes to the baseline plan. In the Adapt Phase, the actual result is monitored for making progress and making changes where necessary and see it as an opportunity to adapt the changes necessary (even if it was a mistake in design or development). The Agile goal is to be able to start with a solid vision and continuously adapt during the sprints continuously to include important changes that will provide value. The questions to ask in this phase are:

1. Is the value, in the form of a releasable product being delivered?
2. Is the quality goal of building a reliable, adaptable product being met?
3. Is the project progressing satisfactorily within acceptable constraints?
4. Is the team adapting efficiently to changes imposed by management, customers, or technology?

### Sprint Retrospective

Upon the completion of all Sprint work products, and ceremonies, the team came together for a Sprint Retrospective in which we evaluated the iteration. We reflected upon iteration and asked ourselves, what went well, what went wrong and what can we do better.

* What went well?
* What went wrong?
* What can we do better?

# Sprint One

### Sprint Planning Meeting

The Sprint Planning Meeting for the first sprint was conducted within a two hour meeting with the Product Owner, Scrum Master and Scrum Team in attendance. The meeting began with the Product Owner going over the highest priority product backlog items. The team reviewed enough stories for approximately two sprints worth of product backlog items. Each of the reviewed stories received an estimated effort of S, M, L, XL and estimation points were calculated based on a scale. Product backlog items with an estimated effort that was greater than XL were broken down and priority was adjusted.

* S = 2 Estimation Points
* M = 4 Estimation Points
* L = 6 Estimation Points
* XL = 8 Estimation Points

Once enough product backlog items were reviewed and prioritized within the product backlog, items were placed into the sprint backlog. Items in the sprint backlog were assigned tasks and each task was given an estimated time for completion. Items were added into the sprint backlog until the estimated time for completion reached 270 hours, which is the available development hours for the sprint. There was a slight modification made to the initial Release Plan as one story that was planned for the first sprint was moved to the second sprint.

At the end of the Sprint Planning Meeting the team had the following work products:

* Top 5 Risks were identified
* A Sprint goal was defined
* A Sprint Backlog was documented

#### Top 5 Risks

1. Development environment and code repository setup and configuration
2. Database encryption
3. Unit testing
4. Integration testing
5. Security Roles

#### Sprint Goal

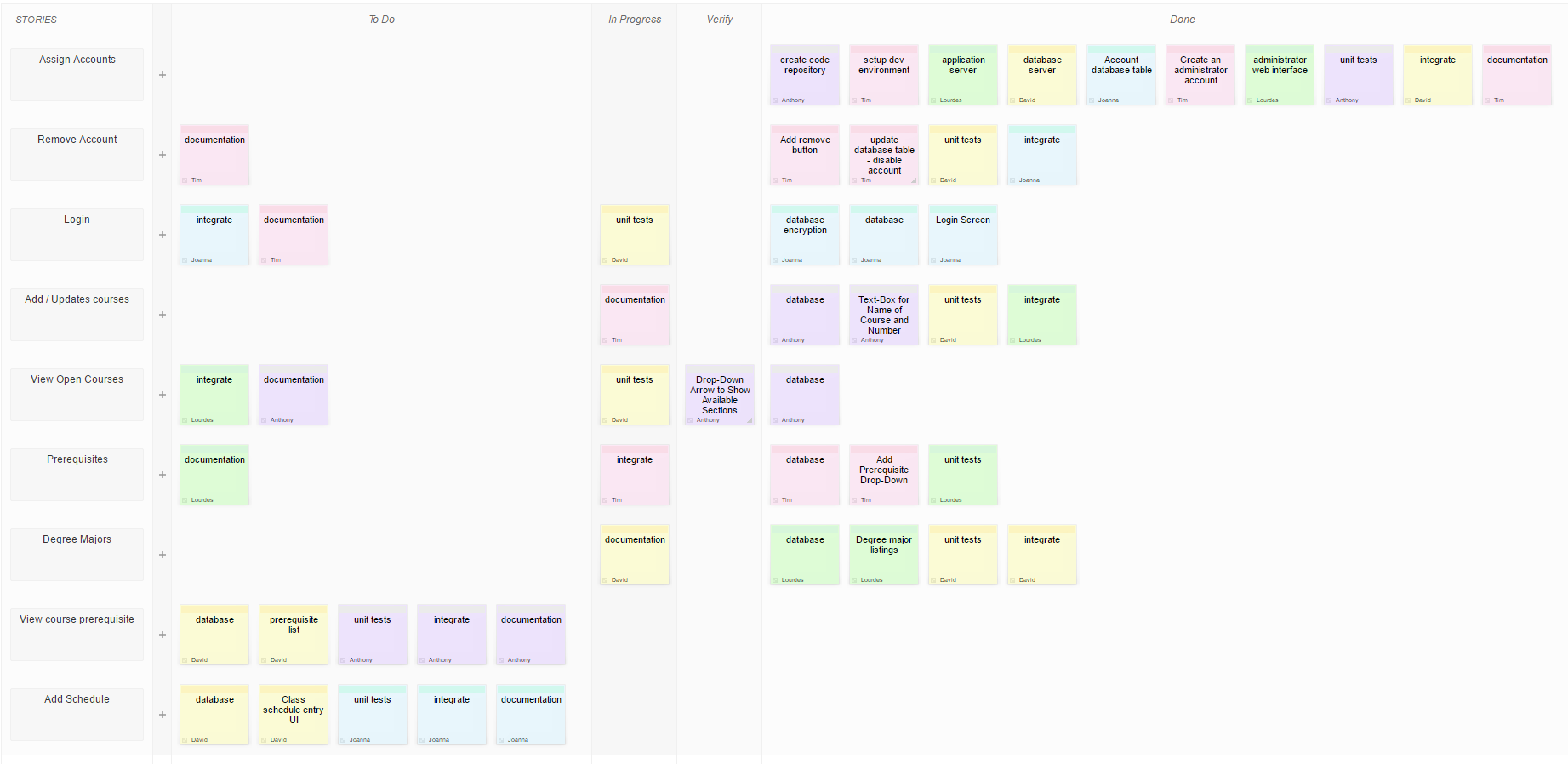
Implement basic schedule functionality for adding, updating and viewing course information. Implement login functionality.

#### Sprint Backlog and Task Board

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Story Id** | **Story Name** | **Story** | **Priority** | **Estimation (S,M, L, XL)** | **Estimation Points** | **Dependencies** | **Acceptance Test** |
| 8 | Assign Accounts | As an administrator, I want to assign students and professors accounts and passwords so that they can log into the system. | 1 | S | 2 |  | 1. The account administrator should be able to assign a temporary password when creating the account  2. There needs to be a way to distinguish between a student account vs a professor account |
| 16 | Remove account | As an administrator, I want the ability to remove a student or professor's authorization so that access is maintained up to date. | 2 | S | 2 |  | 1. Accounts will be set to inactive in order to retain history  2. Password is removed |
| 9 | Login | As a user, I want to be able to log into the system so that I am able to use the system. | 3 | S | 2 |  | 1. User is able to access account when created  2. Authenticate username and password  3. User should be able to reset password if the account is active  4. On initial login, user will have to reset password |
| 19 | Add / Updates courses | As an administrator, I want to be able to enter courses into the system so that they are available to be added to schedules. | 4 | M | 4 |  | 1. Course should contain a course name, link to the course description, number of units.  2. The ability to disable a course |
| 11 | View open courses | As a student, I want to view all open sections of a course so that I can see courses to enroll in. | 5 | S | 2 | 7 | 1. The ability for a student to search or select an open section of a course |
| 4 | Prerequisites | As an administrator, I want to add prerequisites so that students cannot enroll in a class for which he has not taken prerequisite courses. | 6 | M | 4 | 19 | 1. Add prerequisite to existing courses  2. Add prerequisite when adding new course |
| 6 | Degree majors | As an administrator, I want to list the courses for degree majors so that students will know which classes to take to qualify for their major. | 7 | S | 2 |  | 1. Ability to add degree majors  2. Ability to add courses required for each degree major  3. Ability to update degree majors and courses |
| 12 | View course prerequisite | As a student, I want to see the prerequisites for a course so that I can see if I am eligible to enroll in the course. | 8 | S | 2 | 4, 7, 11 | Click on view prerequisites link to display the courses that are prerequisites |
| 7 | Add Schedule | As an administrator, I want to enter the class schedule for a semester so that students will know the availability of classes. | 9 | M | 4 |  | 1. Ability to add time, location, instructor, and number or seat in the course |

[**Sprint 1 Task Board**](https://scrumy.com/faun87elating)

The Task Board was created at the beginning of the sprint and was updated each day during the Daily Scrum Meeting. Tasks were moved from the “To Do” column, to “In Progress”, “Verify”, and “Done”.

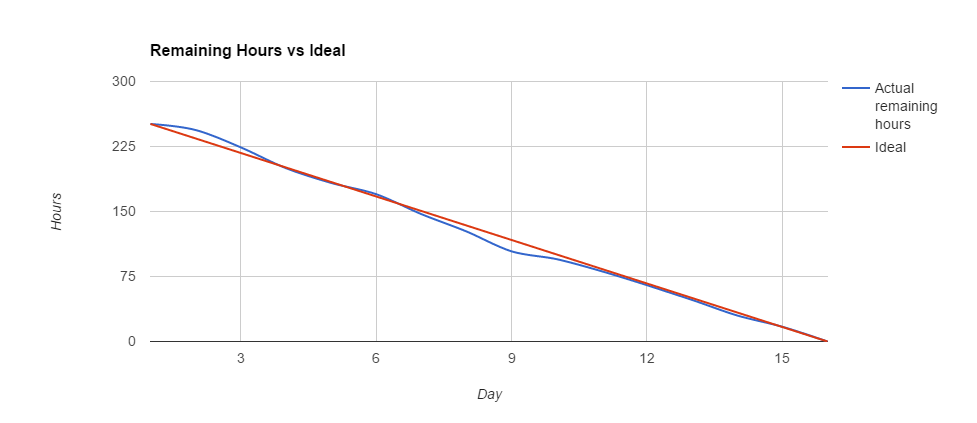


### Issue Tracking

|  |  |  |
| --- | --- | --- |
| **Sprint Day** | **Issue** | **Resolution** |
| Day 1 | None | None |
| Day 2 | Issue creating password database encryption | None |
| Day 3 | None | Encryption issue from Day 2 resolved |
| Day 4 | Authentication of username/password combination; not authenticating | None |
| Day 5 | Day 4 issue | None |
| Day 6 | None | Resolution of authentication Issue |
| Day 7 | None | None |
| Day 8 | User interface not displaying classes correctly, shows an unknown image symbol | None |
| Day 9 | None | Resolved interface issue |
| Day 10 | Major not correctly shown with courses | None |
| Day 11 | Day 10 issue | None |
| Day 12 | None | Resolution of major courses correlation |
| Day 13 | Half of the courses when searched were not displayed correctly | None |
| Day 14 | Day 13 Issue | None |
| Day 15 | None | Resolution of course display. Correctly displays courses |
| Day 16 | None | None |

### Sprint Burn Down Chart

As the tasks were updated each day during the Daily Scrum Meeting, the number of remaining hours was captured each day and documented on the Burn Down chart. The Burn Down chart helped the team visualize the status of the iteration.

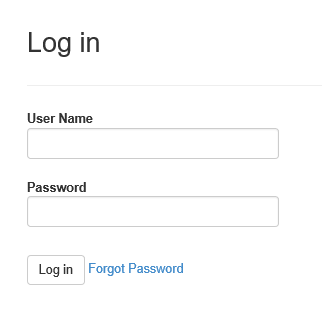


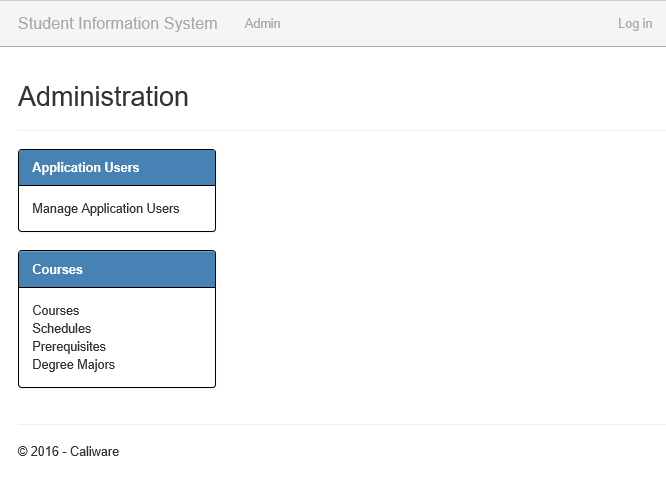
### Retrospective (Adapt Phase)

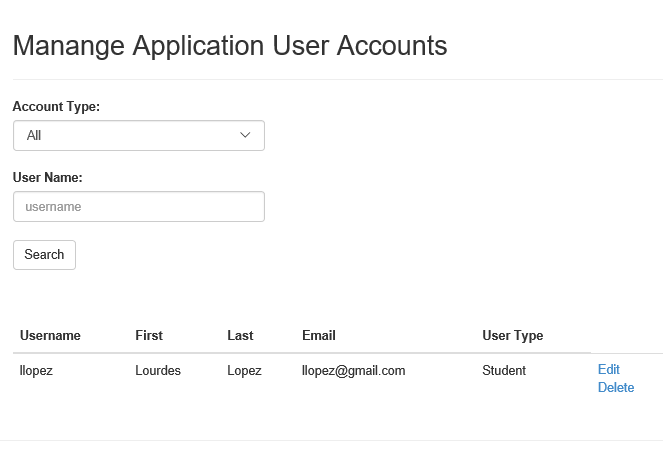
* What went well?
  + Towards the middle of the sprint UI tasks were completed ahead of schedule.
* What went wrong?
  + At the beginning of the iteration the team fell slightly behind schedule due to setup difficulties, encryption, not knowing how to do documentation and complexity with setting up code repository and servers.
  + The team encountered difficulty in setting up the database encryption.
* What can we do better?
  + Add more time for initial research and training for tasks with unfamiliar technology.

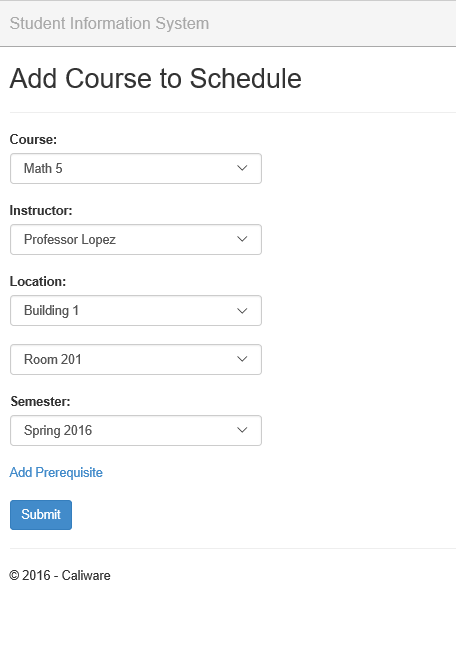
### Sprint Review Meeting

At the end of the first sprint, a Sprint Review Meeting was held. A demo was conducted for the stakeholders that covered features completed within the first sprint. Attendees included the Scrum Team, the Scrum Master, the Product Owner, executives, management and customers.









# Sprint Two

### Sprint Planning (Adapt Phase)

The planning session for the second sprint was conducted similar to the first sprint. During the sprint planning for the second sprint an additional story was added to the product backlog. Story priority was reviewed but remained the same. Acceptance criteria was defined for the stories going into the sprint backlog.

**Additional Story added to the Product Backlog**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Story Id** | **Story Name** | **Story** | **Priority** | **Planned Iteration** | **Estimation (S,M, L, XL)** | **Estimation Points** | **Dependencies** | **Acceptance Test** |
| 21 | Major Progress | As a student, I want the ability to view a list of requirements for my major and an indication if the requirement is completed so that I can know my progress. | 21 | 4 | M | 4 |  |  |

#### Top 5 Risks

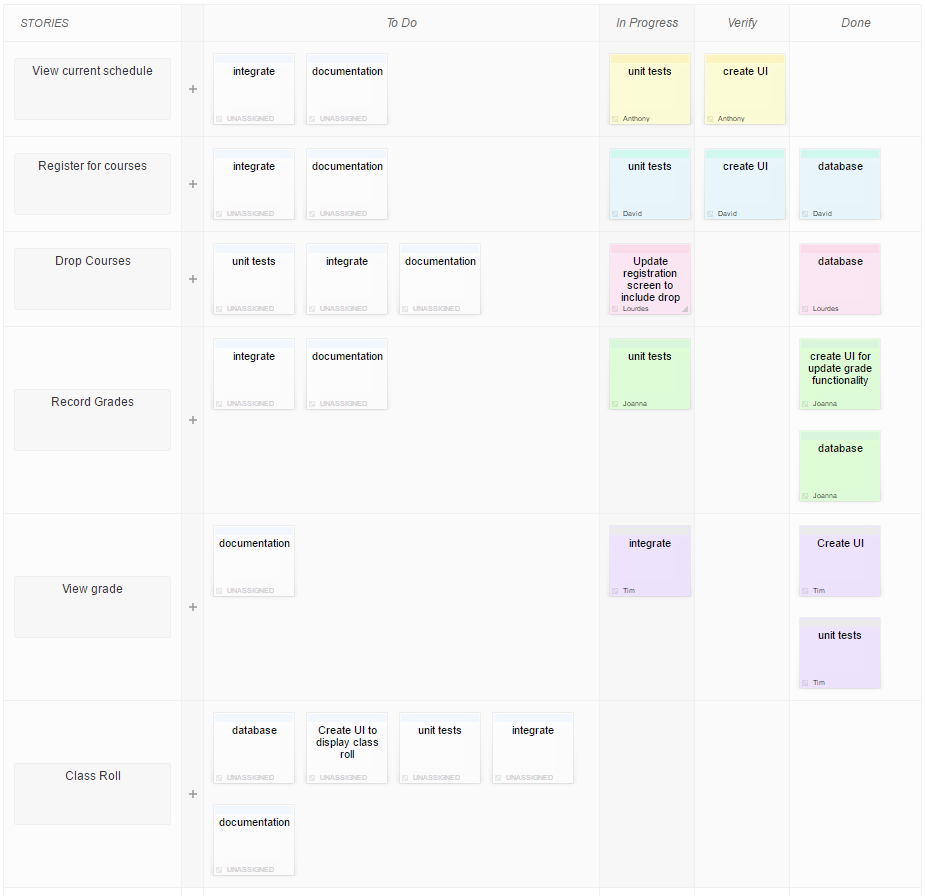
1. Registration functionality
2. Integrating registration with existing course prerequisite functionality
3. Drop courses functionality
4. Record grades functionality
5. Update grades functionality

#### Sprint Goal

Implement registration functionality and allow professor to enter grades.

#### Sprint Backlog and Task Board

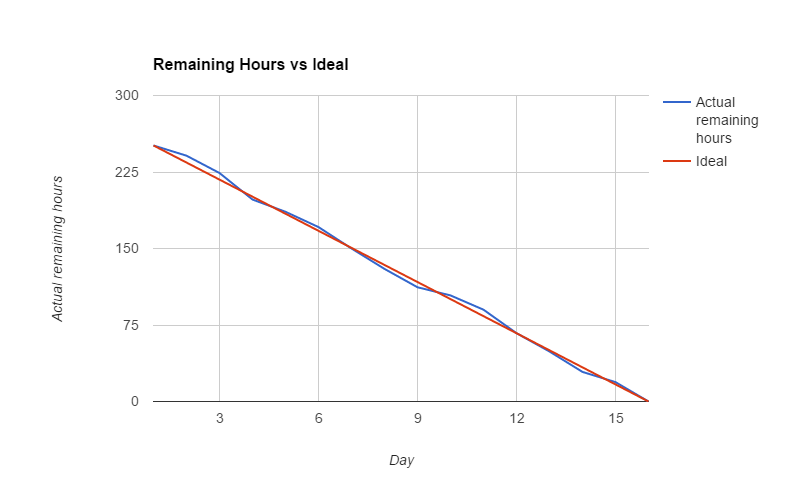
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Story Id | Story Name | Story | Priority | Estimation (S,M, L, XL) | Estimation Points | Dependencies | Acceptance Test |
| 15 | View current schedule | As a student, I want to see my current schedule so that I know what classes I am taking for the semester. | 10 | S | 2 |  | 1. Student schedule shall display as a list and will include date, time, professor and location. |
| 10 | Register for Courses | As a student, I want the ability to register for courses so that I can enroll for the semester. | 11 | XL | 8 |  | 1. Ensure that prerequisite is met.2. Ensure that enrollment capacity is not exceeded. |
| 13 | Drop Course | As a student, I want to be able to drop a course so that I can adjust my enrollment. | 12 | L | 6 |  | 1. Only allow during first 2 weeks of classes. |
| 3 | Record Grades | As a professor, I want the ability to add/update student's grades so that I can keep track of student's progress and status. | 13 | M | 4 |  | 1. Professor should have permissions to enter grades.2. Grades can be entered one student at a time or as a list with all enrolled students.3. Professor cannot update grades after they have been submitted at the end of the semester. |
| 14 | View grade | As a student, I want to see my grades so that I know if I am passing my courses. | 14 | M | 4 |  | 1. Ensure that student can only see their own grade and not the grades for the entire class. |
| 1 | Class Roll | As a professor, I want the ability to view a class roster so that I can know what students are enrolled. | 15 | S | 2 |  | 1. Professor can view the students names currently enrolled in the class. |

[Sprint 2 Task Board](https://scrumy.com/staccati03toil)

### Issue Tracking

|  |  |  |
| --- | --- | --- |
| **Sprint Day** | **Issue** | **Resolution** |
| Day 1 | None | None |
| Day 2 | None | None |
| Day 3 | When trying to register for classes it was not allowing it due to incorrect prerequisites | None |
| Day 4 | None | Resolved prerequisite issue |
| Day 5 | States the class was dropped but did not actually drop course | None |
| Day 6 | Course dropping issue | None |
| Day 7 | Grades were not showing as a letter but symbols | Course dropping issue resolved |
| Day 8 | None | Grade issue resolved |
| Day 9 | Grade changing issue caused issue with adding courses to reoccur as it does not show the prerequisites as passed | None |
| Day 10 | None | Grade is now showing correctly. This resolution fixed prerequisite issue. |
| Day 11 | None | None |
| Day 12 | None | None |
| Day 13 | None | None |
| Day 14 | Class roster showing all classes the professor is currently teaching | None |
| Day 15 | None | Class roster split correctly |
| Day 16 | None | None |

### Sprint Burn Down Chart

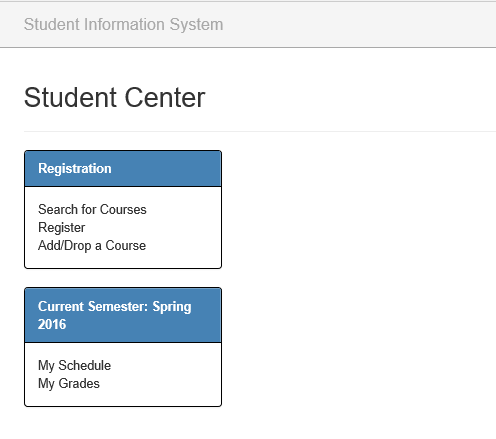


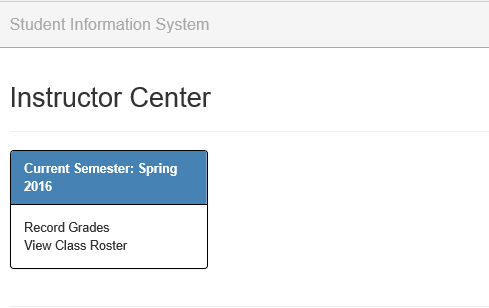
### Retrospective (Adapt Phase)

* What went well?
  + Overall task hours/effort estimates
* What went wrong?
  + Several errors found during integration testing
* What can we do better?
  + More thorough unit tests

### Sprint Review Meeting

At the end of the second sprint, a Sprint Review Meeting was held similar to the first review meeting. A demo was conducted for the stakeholders that covered features completed within the second sprint. Attendees included the Scrum Team, the Scrum Master, the Product Owner, executives, management and customers.





# Sprint Three

### Sprint Planning (Adapt Phase)

The planning session for the third sprint was conducted similar to the first sprint. During the sprint planning for the third sprint an additional story was added to the product backlog. Story priority was reviewed but remained the same. Acceptance criteria was defined for the stories going into the sprint backlog.

**Additional Story Added to the Product Backlog**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Story Id** | **Story Name** | **Story** | **Priority** | **Planned Iteration** | **Estimation (S,M, L, XL)** | **Estimation Points** | **Dependencies** | **Acceptance Test** |
| 22 | Major Progress | As a student, I want the ability to create a plan so that I can keep track of which classes I will be taking in future semesters. | 22 | 4 | M | 4 |  |  |

#### Top 5 Risks

1. Graduation Check functionality
2. Integrating graduation check with major requirements
3. Integrating graduation check with updating grades
4. Student Transcripts
5. Update Grade Level functionality

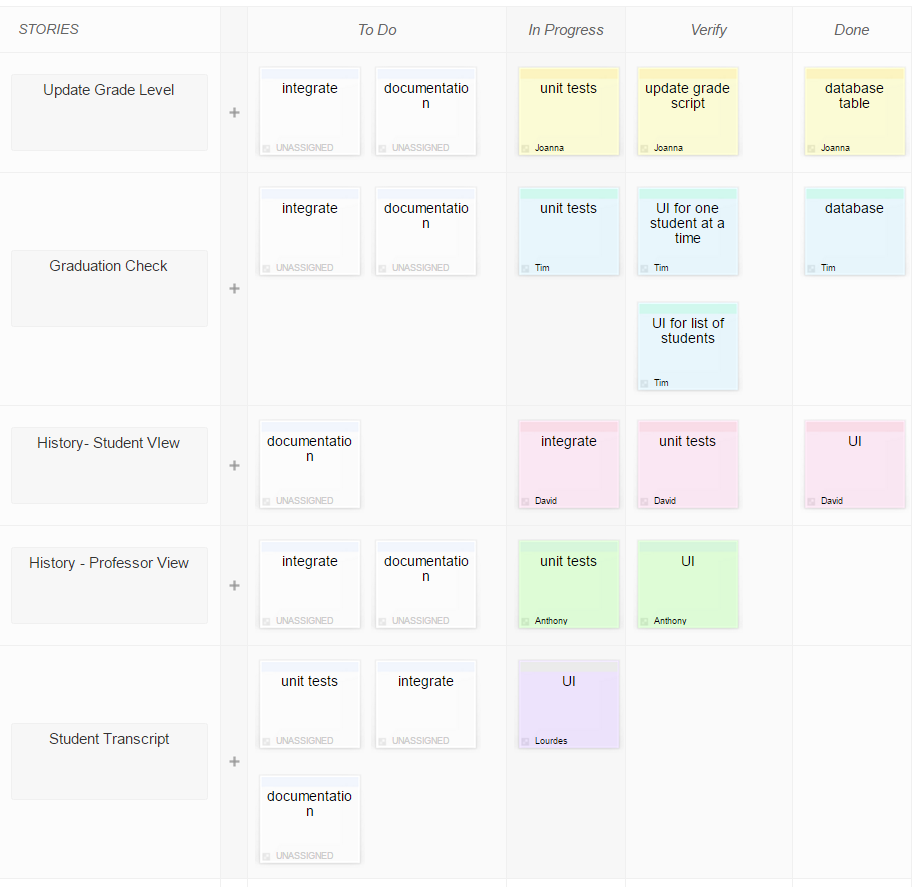
#### Sprint Goal

Implement transcript, graduation check and access to student record functionality.

#### Sprint Backlog and Task Board

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Story Id** | **Story Name** | **Story** | **Priority** | **Estimation (S,M, L, XL)** | **Estimation Points** | **Dependencies** | **Acceptance Test** |
| 5 | Update Grade Level | As an administrator, I want the system to update grade level based on the number of units so the grade level of the students are automatically assigned. | 16 | L | 6 |  | 1. Grade level update should occur upon submission of final grades.  2. Grade level updates should also occur upon correction of final grade. |
| 20 | Graduation Check | As an administrator, I want the system to perform graduation checks so that I can determine if a student has completed the specific courses required for his/her major. | 17 | XL | 8 |  | 1. Graduation check should occur upon UI request.  2. Administrator should have the ability to request a grade check for one student at a time or for a list of students.  3. List can be created by allowing the administrator to search students by grade level. |
| 17 | History | As a student, I want to see my history of courses taken and grades earned so that I can track the courses I have taken. | 18 | S | 2 |  | 1. View only, student history should not be removed. |
| 2 | Student History | As a professor, I want the ability to view a student's course history so that I can see which classes the student has previously taken. | 19 | S | 2 |  | 1. Professor can see a history of courses taken by a student.  2. Professor should be able to access history by searching for an individual student or by a link from the class roll. |
| 18 | Student Transcript | As an administrator, I want the system to produce student transcripts so that I can verify student history, grades, and graduation. | 20 | XL | 8 |  | 1. Student transcript shall be produced upon demand.  2. Student transcripts can be requested by student or administration. |

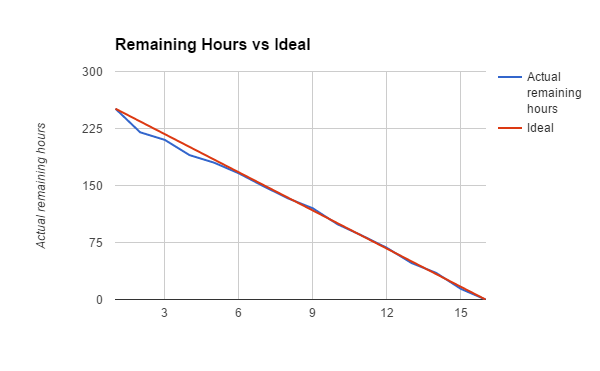
[**Sprint 3 Task Board**](https://scrumy.com/sermon33clammier)



### Issue Tracking

|  |  |  |
| --- | --- | --- |
| **Sprint Day** | **Issue** | **Resolution** |
| Day 1 | Update for students failing after second year changes | None |
| Day 2 | Issue same as day 1 | None |
| Day 3 | None | Update working as intended |
| Day 4 | Graduation allowance granted to students who had five credits under the needed amount | None |
| Day 5 | Graduation now not allowing those who should be allowed to graduate | None |
| Day 6 | None | Graduation requirement now allowing those who should be graduating to show as ready to graduate |
| Day 7 | None | None |
| Day 8 | Student history was only showing four years rather than as many as the student had attended the school | Those who were under graduation requirements are now not permitted to graduate |
| Day 9 | None | None |
| Day 10 | None | Student history shows all years the student has attended the institution |
| Day 11 | Professors can search history by student but not by identifier | None |
| Day 12 | None | None |
| Day 13 | None | Identifiers were not being uploaded to the database correctly |
| Day 14 | Transcripts only showing first four years | Identifiers now allowed for searching history |
| Day 15 | None | Same issue as day 8 and has been resolved |
| Day 16 | None | None |

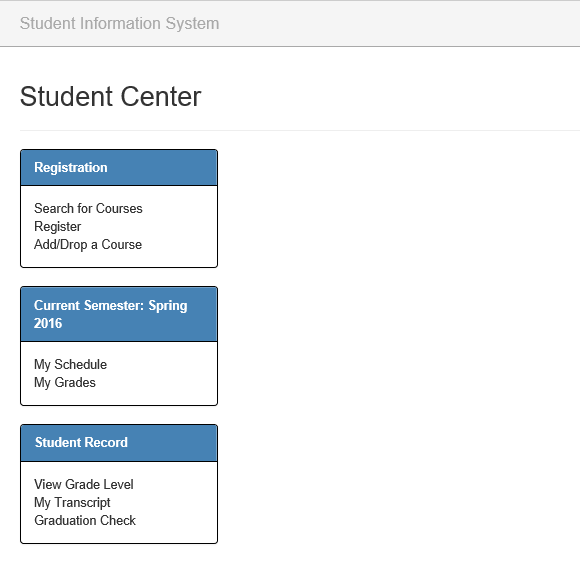
### Sprint Burn Down Chart

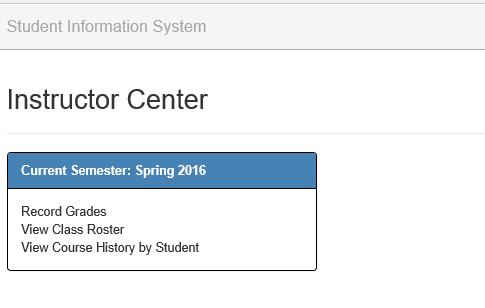


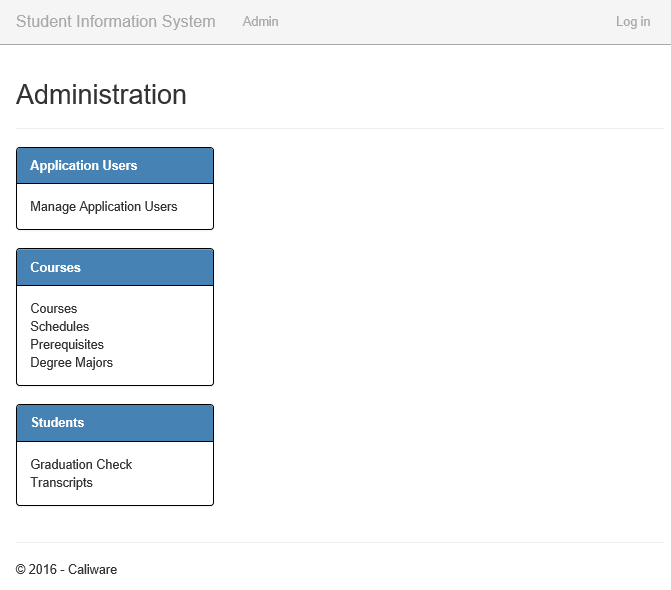
### Retrospective (Adapt Phase)

* What went well?
  + Excellent time management, ahead of schedule for the first part of iteration
* What went wrong?
  + Same issues were being shown multiple times
* What can we do better?
  + Test resolutions more rigorously for future changes

### Sprint Review Meeting







## Final Product Backlog

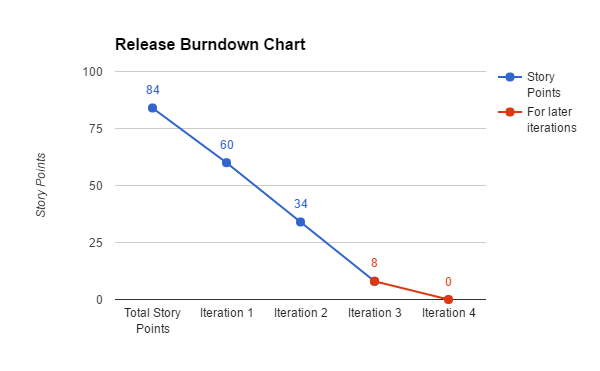
The product backlog was reviewed and adjusted prior to each sprint. Acceptance criteria was added for the stories selected for each sprint during the planning meeting. Product backlog items were prioritized at each sprint.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Story Id** | **Story Name** | **Story** | **Priority** | **Planned Iteration** | **Estimation (S,M, L, XL)** | **Estimation Points** | **Dependencies** | **Acceptance Test** | **Status** |
| 8 | Assign Accounts | As an administrator, I want to assign students and professors accounts and passwords so that they can log into the system. | 1 | 1 | S | 2 |  | 1. The account administrator should be able to assign a temporary password when creating the account  2. There needs to be a way to distinguish between a student account vs a professor account | Completed |
| 16 | Remove account | As an administrator, I want the ability to remove a student or professor's authorization so that access is maintained up to date. | 2 | 1 | S | 2 |  | 1. Accounts will be set to inactive in order to retain history  2. Password is removed | Completed |
| 9 | Login | As a user, I want to be able to log into the system so that I am able to use the system. | 3 | 1 | S | 2 |  | 1. User is able to access account when created  2. Authenticate username and password  3. User should be able to reset password if the account is active  4. On initial login, user will have to reset password | Completed |
| 19 | Add / Updates courses | As an administrator, I want to be able to enter courses into the system so that they are available to be added to schedules. | 4 | 1 | M | 4 |  | 1. Course should contain a course name, link to the course description, number of units.  2. The ability to disable a course | Completed |
| 11 | View open courses | As a student, I want to view all open sections of a course so that I can see courses to enroll in. | 5 | 1 | S | 2 | 7 | 1. The ability for a student to search or select an open section of a course | Completed |
| 4 | Prerequisites | As an administrator, I want to add prerequisites so that students cannot enroll in a class for which he has not taken prerequisite courses. | 6 | 1 | M | 4 | 19 | 1. Add prerequisite to existing courses  2. Add prerequisite when adding new course | Completed |
| 6 | Degree majors | As an administrator, I want to list the courses for degree majors so that students will know which classes to take to qualify for their major. | 7 | 1 | S | 2 |  | 1. Ability to add degree majors  2. Ability to add courses required for each degree major  3. Ability to update degree majors and courses | Completed |
| 12 | View course prerequisite | As a student, I want to see the prerequisites for a course so that I can see if I am eligible to enroll in the course. | 8 | 1 | S | 2 | 4, 7, 11 | Click on view prerequisites link to display the courses that are prerequisites | Completed |
| 7 | Add Schedule | As an administrator, I want to enter the class schedule for a semester so that students will know the availability of classes. | 9 | 1 | M | 4 |  | 1. Ability to add time, location, instructor, and number or seat in the course | Completed |
| 15 | View current schedule | As a student, I want to see my current schedule so that I know what classes I am taking for the semester. | 10 | 2 | S | 2 |  | 1. Student schedule shall display as a list and will include date, time, professor and location. | Completed |
| 10 | Register for Courses | As a student, I want the ability to register for courses so that I can enroll for the semester. | 11 | 2 | XL | 8 |  | 1. Ensure that prerequisite is met.  2. Ensure that enrollment capacity is not exceeded. | Completed |
| 13 | Drop Course | As a student, I want to be able to drop a course so that I can adjust my enrollment. | 12 | 2 | L | 6 |  | 1. Only allow during first 2 weeks of classes. | Completed |
| 3 | Record Grades | As a professor, I want the ability to add/update student's grades so that I can keep track of student's progress and status. | 13 | 2 | M | 4 |  | 1. Professor should have permissions to enter grades.  2. Grades can be entered one student at a time or as a list with all enrolled students.  3. Professor cannot update grades after they have been submitted at the end of the semester. | Completed |
| 14 | View grade | As a student, I want to see my grades so that I know if I am passing my courses. | 14 | 2 | M | 4 |  | 1. Ensure that student can only see their own grade and not the grades for the entire class. | Completed |
| 1 | Class Roll | As a professor, I want the ability to view a class roster so that I can know what students are enrolled. | 15 | 2 | S | 2 |  | 1. Professor can view the students names currently enrolled in the class. | Completed |
| 5 | Update Grade Level | As an administrator, I want the system to update grade level based on the number of units so the grade level of the students are automatically assigned. | 16 | 3 | L | 6 |  | 1. Grade level update should occur upon submission of final grades.  2. Grade level updates should also occur upon correction of final grade. | Completed |
| 20 | Graduation Check | As an administrator, I want the system to perform graduation checks so that I can determine if a student has completed the specific courses required for his/her major. | 17 | 3 | XL | 8 |  | 1. Graduation check should occur upon UI request.  2. Administrator should have the ability to request a grade check for one student at a time or for a list of students.  3. List can be created by allowing the administrator to search students by grade level. | Completed |
| 17 | History | As a student, I want to see my history of courses taken and grades earned so that I can track the courses I have taken. | 18 | 3 | S | 2 |  | 1. View only, student history should not be removed. | Completed |
| 2 | Student History | As a professor, I want the ability to view a student's course history so that I can see which classes the student has previously taken. | 19 | 3 | S | 2 |  | 1. Professor can see a history of courses taken by a student.  2. Professor should be able to access history by searching for an individual student or by a link from the class roll. | Completed |
| 18 | Student Transcript | As an administrator, I want the system to produce student transcripts so that I can verify student history, grades, and graduation. | 20 | 3 | XL | 8 |  | 1. Student transcript shall be produced upon demand.  2. Student transcripts can be requested by student or administration. | Completed |
| 21 | Student Major Progress | As a student, I want the ability to view a list of requirements for my major and an indication if the requirement is completed so that I can know my progress. | 21 | 4 | M | 4 |  |  |  |
| 22 | Student Planner | As a student, I want the ability to create a plan so that I can keep track of which classes I will be taking in future semesters. | 22 | 4 | M | 4 | 21 |  |  |

## Final Release Plan

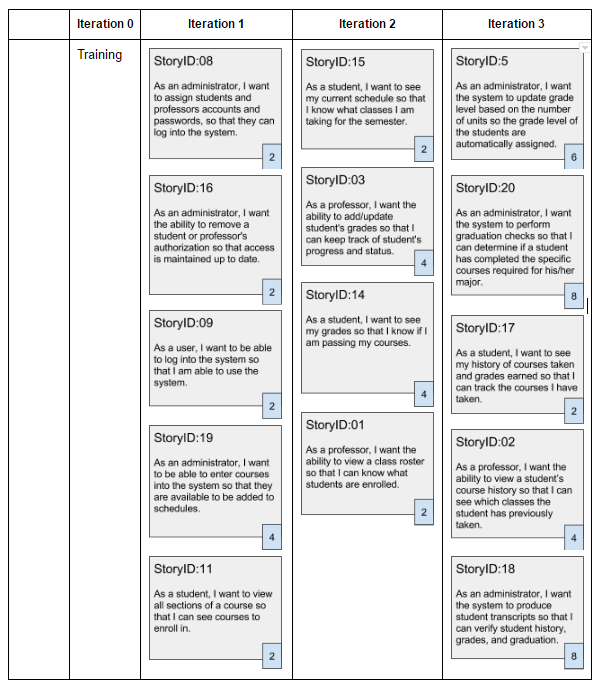
The Release Plan was reviewed and updated prior to each sprint. During the planning meeting for sprint one, a story was moved from sprint one to sprint two. During each of the planning meetings for sprints two and three, a story was added and planned for future sprints.

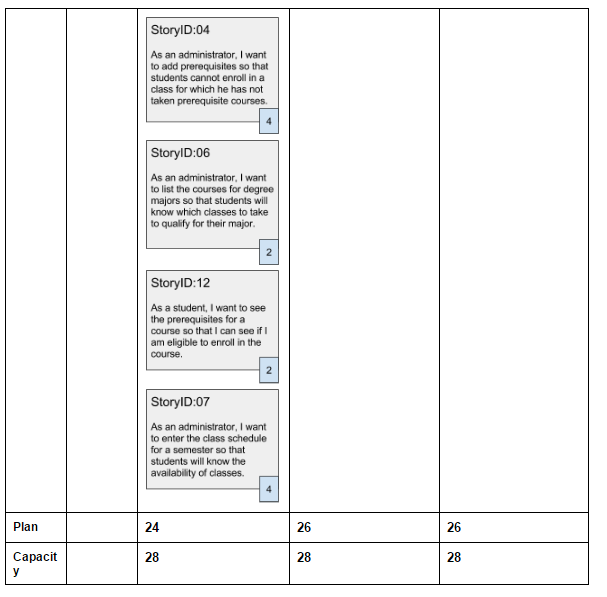
### Release Burndown Chart



### Velocity

Our historical average is 26-28 story points per sprint.





# Close Phase

Projects are finite and it is an important aspect of project management to officially end the project and give well deserved recognition to the team members. The commitments in the product backlog in the form of User Stories have been moved to the Done column. It is up to the Product Owner’s discretion if some incomplete or not started User Stories can be removed from the product backlog. It is not advisable for projects for a given product to be ongoing because without an official close to the project, the team will not do any close processes and it will add to the technical debt. Upon the completion of all Sprints we conducted a Sprint Retrospective in which we evaluated the iteration and asked ourselves, what went well, what went wrong, and what can we do better. After the project overall project was completed we conducted the Project Retrospective. We reflected upon what was learned during the sprints and how we can improve our approach in future projects.

### Project Retrospective

* What went well?
  + Estimates were consistently on target throughout the project
* What went wrong?
  + Towards the beginning of the project there were some complexities with encryption
* What can we do better?
  + Add more time for initial research and training for stories with unfamiliar technology

# Our Process vs CMMI Project Monitoring Control and Project Planning

Capability Maturity Model Integration (CMMI) is a collection of best practices for organizations in software development. Project Monitoring and Control (PMC) and Project Planning (PP) are both under Project Management, which enables organizations to increase efficiency and effectiveness in completing project objectives. Specifically, PMC monitors the project process and PP establishes and maintain plans for project activities. PMC and PP are indicative of organizations with a Maturity Level (ML) 2, a process area that focuses on change and project management. Scrum, a pre-defined development lifecycle based on agile principles, shows some Maturity Level 2 practices, and these Scrum practices can be represented in the specific goals and practices of PMC and PP.

## Project Monitoring and Control

In CMMI, the purpose of Project Monitoring and Control (PMC) is the “to provide an understanding of the project’s progress so that appropriate corrective actions can be taken when the project’s performance deviates significantly from the plan” (CMMI Product Team, 2010).

|  |  |
| --- | --- |
| **CMMI v 1.3 Project Monitoring and Control** | **Comparison to Scrum Practices** |
| **SG 1 Monitor the Project Against the Plan**  *Actual project performance and progress are monitored against the project plan* |  |
| **SP 1.1 Monitor Project Planning Parameters**  *Monitor actual values of project planning parameters against the project plan.* | Burn Down chart  Sprint Backlog  Product Backlog |
| **SP 1.2 Monitor Commitments**  *Monitor commitments against those identified in the project plan.* | Daily Scrum Meeting  Sprint Review Meeting  Burn Down chart |
| **SP 1.3 Monitor Project Risks**  *Monitor risks against those risks identified in the project plan.* | Top 5 risks are identified before each Sprint |
| **SP 1.4 Monitor Data Management**  *Monitor the management of project data against the project plan.* |  |
| **SP 1.5 Monitor Stakeholder Involvement**  *Monitor stakeholder involvement against the project plan.* | Daily Scrum Meeting  Sprint Review Meeting |
| **SP 1.6 Conduct Progress Reviews**  *Periodically review the project’s progress, performance, and issues* | Daily Scrum Meeting  Sprint Review Meeting  Retrospectives |
| **SP 1.7 Conduct Milestone Reviews**  *Review the project’s accomplishments and results at selected project milestones* | Sprint Review Meeting |
| **SG 2 Manage Corrective Action to Closure**  *Corrective actions are managed to closure when the project’s performance or results deviate significantly from the plan.* |  |
| **SP 2.1 Analyze Issues**  *Collect and analyze issues and determine corrective actions to address them.* | Daily Scrum Meeting  Sprint Review Meeting  Sprint Retrospective Meeting |
| **SP 2.2 Take Corrective Action**  *Take corrective action on identified issues.* | Daily Scrum Meeting  Sprint Review Meeting  Sprint Retrospective Meeting |
| **SP 2.3 Manage Corrective Actions**  *Manage corrective actions to closure.* | Daily Scrum Meeting  Sprint Review Meeting |

## Project Planning

In CCMI-DEV, the purpose of Project Planning is to “establish and maintain plans that define project activities” (CMMI Product Team, 2010). Project Planning involves developing the project plan, interacting with stakeholders, obtaining commitment to the plain, and maintaining the plan (Kolb, 2005). Project Planning begins with requirements that define the product and project. The table below shows Scrum practices in comparison to the requirements defined in Project Planning Specific Goals (SG) and Specific Practices (SP).

|  |  |
| --- | --- |
| **CMMI v 1.3 Project Planning** | **Comparison to Scrum Practices** |
| **SG 1 Establish Estimates**  *Estimates of project planning parameters are established and maintained.* |  |
| **SP 1.1 Estimate the Scope of the Project**  *Establish a top-level work breakdown structure (WBS) to estimate the scope of the project.* | Pre-Game Planning phase  Product Backlog |
| **SP 1.2 Establish Estimates of Work Product and Task Attributes**  *Establish and maintain estimates of work product and task attributes.* | * Uses estimation cards of Small, Medium, Large or Extra Large * Estimation points for each story * Establish Acceptance Criteria |
| **SP 1.3 Define Project Lifecycle Phases**  *Define project lifecycle phases on which to scope the planning effort.* | The Scrum lifecycle process   * Planning * Staging * Development * Release |
| **SP 1.4 Estimate Effort and Cost**  *Estimate the project’s effort and cost for work products and tasks based on estimation rationale.* | Ideal time estimates   * Product backlog (high level estimates with estimates cards ) * Sprint backlog (more detailed as defined by the team in estimation of hours) |
| **SG 2 Develop a Project Plan**  *A project plan is established and maintained as the basis for managing the project.* |  |
| **SP 2.1 Establish the Budget and Schedule**  *Establish and maintain the project’s budget and schedule.* | Estimation points  Ideal time estimates  Product Backlog  Sprint Backlog  Scheduled is divided into 30-day Sprints. |
| **SP 2.2 Identify Project Risks**  *Identify and analyze project risks.* | Top 5 risks are identified before each Sprint |
| **SP 2.3 Plan Data Management**  *Plan for the management of project data.* |  |
| **SP 2.4 Plan the Project’s Resources**  *Plan for resources to perform the project.* | Estimation points  Ideal time estimates  Release Plan  Sprint Backlog |
| **SP 2.5 Plan Needed Knowledge and Skills**  *Plan for knowledge and skills needed to perform the project.* |  |
| **SP 2.6 Plan Stakeholder Involvement**  *Plan the involvement of identified stakeholders.* | **Scrum Roles:**  **Scrum Team**   * Cross-functional group * Attempts to build a potentially shippable product increment every Sprint * Collaborates * Self-organizing   **Scrum Master**   * Has no management authority * Doesn't have a project manager role   + Split between Product Owner and team; master acts as a facilitator * Facilitator * Protects team from distractions and interruptions   + Removes impediments   + Facilitates the process   + Helps teach people how to use Scrum   + Promotes improved engineering practices   + Forces time-boxes   + Promotes visibility   **Product Owner**   * Responsible for ROI * Final arbiter of requirements questions * Focused more on the what than on how * Important on vision * Makes business decisions * Prioritizes |
| **SP 2.7 Establish the Project Plan**  *Establish and maintain the overall project plan.* | Release Plan  Product backlog  Sprint Backlog |
| **SG 3 Obtain Commitment to the Plan**  *Commitments to the project plan are established and maintained.* |  |
| **SP 3.1 Review Plans That Affect the Project**  *Review all plans that affect the project to understand project commitments.* | Sprint Planning Meeting  Daily Scrum Meeting |
| **SP 3.2 Reconcile Work and Resource Levels**  *Adjust the project plan to reconcile available and estimated resources.* | Sprint Planning Meeting  Daily Scrum Meeting |
| **SP 3.3 Obtain Plan Commitment**  *Obtain commitment from relevant stakeholders responsible for performing and supporting plan execution.* | Backlog Refinement meeting  Sprint Planning Meeting  Daily Scrum Meeting |

# References

Beck, K., Beedle, M., Bennekum, A. V., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., Thomas, D. (2001). Manifesto for Agile Software Development.

CMMI Product Team. (2010, November). CMMI® for development, version 1.3. *Software Engineering Institute*. Retrieved from <http://www.sei.cmu.edu/reports/10tr033.pdf>

Kolb, P. (2005, May). CMMI: Specific goals and practices. *ETH Software Engineering*. Retrieved from <http://se.inf.ethz.ch/old/teaching/ws2005/0273/slides/CMMI%20SG%20and%20S%20Practices.pdf>

Scrummy [Computer software]. (n.d.). Retrieved May 2016, from [https://Scrumy.com/](https://scrumy.com/)