

Sprint 3 MVP Report

1. Deployed App

Link: <https://ea-block-scheduler-4fec886e389.herokuapp.com/generate-schedule>

2. Sprint Goal

By the end of this sprint, we will have fully developed admin and user dashboards, controllers to allow users the ability to login using single sign on, a completed page of available classes with their prerequisites, and a parser for excel documents.

#34 Add SSO Capability - [x] Basic setup with third party

#33 Add tokens for logging in - [x] Add session logic for logged-in users

#35 Connect login page to student dashboard - [x] Add authorization elements to dashboard controller components

#78 Sprint documentation - [x] Sprint 3 documentation

#17 Upload service for classes spreadsheet - [x] Create page for uploading class spreadsheet to data - [x] Create parsing service for backend data persistence

#19 Store the spreadsheet data on the database - [x] Add styling to the upload excel page

#75 Create session and user table - [x] Create session and user table

#37 Set block generating conditions on the web - [x] Block generating conditions are made

#61 Display Class Details UI - [x] Display Class Details: 2 hours (Chengyan Tsai)

3. Sprint Backlog

#38 application makes possible blocks with conditions - [] Develop algorithm to apply block generating conditions for typical blocks made by admin

#40 user pick rest of classes - [] Create pop up alerts the user that “scheduled blocks are limited!”

#34 Add SSO Capability - [] Add settings to configs

#19 Store the spreadsheet data on the database - [] Create Database Schema

#36 Set typical classes grouping on the web - [] Create Admin sets typical classes to generate draft blocks on the web page

#39 Generate possible blocks - [] Develop blocks are generated automatically through algorithm

#60 Display Class Blocks UI - [] Display Class Blocks

#11 Show basic schedule timetable UI - [] Create basic time table - [] Create Page

4. Use of BDD

All stories have Cucumber feature(s) defined with several scenarios each, all Cucumber scenarios are passing and all steps are defined

5. Use of TDD

All models and controllers have specs with at least one example/expectation, all examples are passing

6. Test Coverage

Test coverage generated by SimpleCov is at least 90%. > Run `bundle exec cucumber` and `rspec` to see the result

7. Code quality

Code Climate Report

8. Code Style

All Rubocop cops are enabled and report at most 1 style offense per file. > Run `rubocop` to see the result

9. Use of Project Tracker

Taiga User Stories: <https://tree.taiga.io/project/aaronjones05-block-scheduler/taskboard/ecstatic-woodchuck>

10. Presentation

See the deployed app.