

Canvas API

Overview of useful Canvas API endpoints for the back-end of the analytics app.

Name	URL	Description	Role in application	Screenshots
Quiz statistics	https://canvas.instructure.com/doc/api/quiz_statistics.html	Calculating mean/median for entire student cohort	Quiz graphs, quiz completed statistics	<p>A QuizStatisticsSubmissionStatistics object looks like:</p> <pre>// Generic statistics for all submissions for a quiz. { // The number of students who have taken the quiz. "unique_count": 3, // The mean of the student submission scores. "score_average": 4.33333333333333, // The highest submission score. "score_high": 6, // The lowest submission score. "score_low": 3, // Standard deviation of the submission scores. "score_stdev": 1.24721912892465, // A percentile distribution of the student scores, each key is the percentile // (ranges between 0 and 100%) while the value is the number of students who // received that score. "scores": {"50":1,"34":5,"100":1}, // The mean of the number of questions answered correctly by each student. "correct_count_average": 3.66666666666667, // The mean of the number of questions answered incorrectly by each student. "incorrect_count_average": 5, // The average time spent by students while taking the quiz. "duration_average": 42.33333333 }</pre>
Live Events	https://canvas.instructure.com/doc/api/file.data_service_introduction.html	Tracking clicks on particular links /buttons for lectures /tutorials /modules	Number of clicks on lectures /tutorials /modules links	-
Assignment score statistics	https://canvas.instructure.com/doc/api/assignments.html	Calculate assignment scores for individual students	Assignment statistics for the student, assignment graph	<p>A ScoreStatistic object looks like:</p> <pre>// Used by Assignment model { // Min score "min": 1, // Max score "max": 10, // Mean score "mean": 6 }</pre>
Module - completion requirement	https://canvas.instructure.com/doc/api/modules.html	Determine the number of tasks completed (i.e. green tick in the second image)	Lectures /tutorials /modules completed	<p>A CompletionRequirement object looks like:</p> <pre>{ // one of 'must_view', 'must_submit', 'must_contribute', 'min_score', // 'must_mark_done' "type": "min_score", // minimum score required to complete (only present when type == 'min_score') "min_score": 10, // whether the calling user has met this requirement (Optional; present only if // the caller is a student or if the optional parameter 'student_id' is // included) "completed": true }</pre> <div><div>▼ Projects</div><div>Complete All Items</div></div> <div>Teams, Project and Supervisors</div> <div>Project Details</div> <div><div>Creative Learning Units</div><div>Marked as done</div></div> <div><div>Genyus Network - Accessible Interface</div><div>Marked as done</div></div> <div><div>Genyus Network - Facebook Data Migration</div><div>Mark as done</div></div> <div><div>Berry Street</div><div>Mark as done</div></div>

Analytics - course-level assignment data	https://canvas.instructure.com/doc/api/analytcs.html	Get course level assignment data - to show subject averages/top quartile etc.	Assignment graphs (for group statistics)	<p>Example Request:</p> <pre>curl https://<canvas>/api/v1/courses/<course_id>/analytics/assignments \ -H 'Authorization: Bearer <token>'</pre> <p>Example Response:</p> <pre>{ { "assignment_id": 1234, "title": "Assignment 1", "points_possible": 10, "due_at": "2012-01-25T22:00:00-07:00", "unlock_at": "2012-01-20T22:00:00-07:00", "muted": false, "min_score": 2, "max_score": 10, "median": 7, "first_quartile": 4, "third_quartile": 8, "tardiness_breakdown": { "on_time": 0.75, "missing": 0.1, "late": 0.15 } }, { "assignment_id": 1235, "title": "Assignment 2", "points_possible": 15, "due_at": "2012-01-26T22:00:00-07:00", "unlock_at": null, "muted": true, "min_score": 8, "max_score": 8, "median": 8, "first_quartile": 8, "third_quartile": 8, "tardiness_breakdown": { "on_time": 0.65, "missing": 0.12, "late": 0.23 }, "total": 275 } }</pre>
Analytics - user in a course level assignment data	https://canvas.instructure.com/doc/api/analytcs.html	Returns a list of assignments for the course sorted by due date. For each assignment returns basic assignment information, the grade breakdown (including the student's actual grade), and the basic submission information for the student's submission if it exists.	Assignment statistics for the student	<p>Example Request:</p> <pre>curl https://<canvas>/api/v1/courses/<course_id>/analytics/users/<user_id>/assignments \ -H 'Authorization: Bearer <token>'</pre> <p>Example Response:</p> <pre>{ { "assignment_id": 1234, "title": "Assignment 1", "points_possible": 10, "due_at": "2012-01-25T22:00:00-07:00", "unlock_at": "2012-01-20T22:00:00-07:00", "muted": false, "min_score": 2, "max_score": 10, "median": 7, "first_quartile": 4, "third_quartile": 8, "module_ids": [1, 2], "submission": { "posted_at": "2012-01-23T20:00:00-07:00", "submitted_at": "2012-01-22T22:00:00-07:00", "score": 10 } }, { "assignment_id": 1235, "title": "Assignment 2", "points_possible": 15, "due_at": "2012-01-26T22:00:00-07:00", "unlock_at": null, "muted": true, "min_score": 8, "max_score": 8, "median": 8, "first_quartile": 8, "third_quartile": 8, "module_ids": [1], "submission": { "posted_at": null, "submitted_at": "2012-01-22T22:00:00-07:00" } } }</pre>
User - page views	https://canvas.instructure.com/doc/api/users.html	Amount of time user spends on a page	Time spent on a page, lecture minutes watched	<p>A PageView object looks like:</p> <pre>// The record of a user page view access in Canvas { // A UUID representing the page view. This is also the unique request id "id": "1a246780-e085-0130-51de-0a33aa501ef", // If the request is from an API request, the app that generated the access // token "app_name": "Canvas for iOS", // The URL requested "url": "https://canvas.instructure.com/conversations", // The type of context for the request "context_type": "Course", // The type of asset in the context for the request, if any "asset_type": "Discussion", // The rails controller that handled the request "controller": "discussions", // The rails action that handled the request "action": "index", // This field is deprecated, and will always be false "contributed": false, // An approximation of how long the user spent on the page, in seconds "interaction_seconds": 7.21, // When the request was made "created_at": "2013-10-01T19:49:47Z", // A flag indicating whether the request was user-initiated, or automatic (such // as an AJAX call) "user_request": true, // How long the response took to render, in seconds "render_time": 0.369, // The user-agent of the browser or program that made the request "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_0_5) AppleWebKit/536.30.1 (KHTML, like Gecko) Version/6.0.5 Safari/536.30.1", // True if the request counted as participating, such as submitting homework "participated": false, // The HTTP method such as GET or POST "http_method": "GET", // The origin IP address of the request "remote_ip": "173.194.46.71", // The page view links to define the relationships "links": {"user":1234,"account":1234} }</pre>