Assessments in the Remote Environment

While there are challenges when assessing student work in the remote environment, there are advantages as well: improved accessibility, improved flexibility, and the ability to provide responsive, detailed feedback in a variety of formats. Here we present some considerations as you revise and develop assessments for your courses.

Determining Assessment Needs

Preparation of assessments starts with the course learning outcomes and evaluation scheme in a course syllabus.

An assessment must align to one or more well-constructed course learning outcomes. Assignments that do not align can be removed from the course without impacting student learning.

The course design will help determine when, how often, and by what means learning is assessed – whether course concepts stand alone and can have separate assessments or are cumulative and requires assessments as checkpoints on progress.

Selecting Assessment Types

Consider the types of assessments that best align with your course learning outcomes. Use a pedagogical wheel to help determine the types of assessments, and if summative or formative assessments better support your outcomes.

Summative Assessment | Determines degree to which educational goals are met. Assesses what students know, value and what they can do. Assesses knowledge acquisition, application, or transferrable skills. Assesses evidence of a product/process to determine if outcomes are achieved. Often perceived as higher stakes to students.

Formative Assessment | Provides feedback more frequently to help students continually improve their learning. Informs students regarding their achievement of learning outcomes, and their progress as they work towards achieving learning outcomes. Often perceived as lower stakes to students.

General Rule of Thumb: Aim for a balance between formative and summative assessments, with marks weighted towards the summative assessment (e.g., major assignments or exams) to avoid students focusing on “learning for the test”.1
Deciding the Number and Timing

What is the best balance between formative (and frequent assessments) and (higher-stakes) summative assessments?

Benefits of frequent assessment:
- Encourages students to learn on an ongoing basis\(^2\)
- Allows students to get feedback on their learning\(^3\)
- Increases academic achievement\(^4,5\)

Challenges of frequent assessment:
- Increases assessment workload and student anxiety\(^6\)
- If summative, may have adverse effects on academic achievement\(^7\)

Use a [workload estimator tool](#) to estimate how much student effort assessments require with your course activities.

Determining the correct number of assessments is a blend of need, type, and course design. Consider one assessment a week as a benchmark that balances frequency and workload during remote instruction. If more are required, we suggest using strategies to reduce students’ perceived workload and stress.

Reducing Perceived Workload and Stress Levels

Decrease Number of Stressors:
- Keep student workload < 10 hrs/week. Note that media workload is often twice as much as non-media workload (e.g.: a 30-minute video = one hour of work)\(^8,9\)
- Teach students to use necessary technology for assessments at start of course\(^10\)
- Provide clear descriptions of assessments including types, number, timing, instructions, rubrics, and expectations\(^11\)
- Use various assessment types and choice of assessments when possible\(^12\)
- Discuss efforts to keep fairness and academic integrity\(^13\)
- Design assessments to be open-book / open-web\(^14\)

Increase Ability to Cope with Stress:
- Create opportunities for students to build positive relationships with peers and instructors\(^15,16\)
- Allow space for student questions during assessments\(^10\)
- Discuss contingency plans with students regarding technical issues that may arise during assignments\(^17\)

Further questions about assessments?

Contact CITL’s Support Centre
[citl.mun.ca/support](http://citl.mun.ca/support)
Assessments: Balancing learning, feedback and workload

References


