Woodbury University School of Architecture

Criticism 5: Degree Project Research

ARCH 441

Semester: Fall 2021

Time: M/W 10:30-11:45 AM

Location: MH 204

Instructors: Mark Ericson

Office Hours: Tuesday and Friday 1-3pm

Catalog Description

Theory and techniques for analyzing and integrating design methodologies, client or user needs, and site conditions into criteria for preparing for an architectural project are studied. The theoretical and practical context for the Degree Project is researched and developed. Along with the completion of a substantiated written position of intent, a project site is selected, program written, and design methodology articulated. Lecture. Prerequisites: ARCH 341, Criticism Three: Theory of Architecture.

Three Unit Lecture.

Learning Outcomes

Minimum to NAAB Criteria:

PC.5 Research and Innovation : How the program prepares students to engage and participate in architectural research to test and evaluate innovations in the field.

Learning Outcome: Compose a degree project proposal based on a body of architectural research that analyzes form and tectonics while employing rigorous arguments and methods to support a well-defined hypothesis **.**

WSCUC Core Competency: Written Communication

Course Learning Outcomes:

Critical Position: Students will establish a critical position within the discipline of the Architecture.

Research: Students will develop ability to examine sources beyond the course bibliography and integrate them into the development of a project proposal.

Precision: Students will construct a precise and functioning software plug-in. This will be demonstrated in clear code hierarchy, creative use of syntax, transparent commenting, and program functionality.

Plugins for Architecture

> It [a computer] is a medium that can dynamically simulate the details of any other medium, including media that cannot exist physically. It is not a tool, though it can act like many tools. It is the first metamedium, and as such has degrees of freedom for representation and expression never before encountered and as yet barely Investigated. (Alan Kay, "Computer Software", 1984)

Within much of architectural education and practice computers and by extension software are considered tools. Students learn to use "digital tools" to produce images of buildings as plans, sections, elevations, and renderings. Similarly, three-dimensional objects are produced through 3D printing or other means of digital fabrication. When we call a computer or a piece of software a tool, we do so with the intent of carefully separating the work of the tool from the intent of the designer/author. The tool is deployed by the designer to produce a specific result. The tool is little more than a facilitator. The tool is not the author. At least this is the argument.

There is however another way to approach the problem of software in Architecture. It is to engage software as a medium in its own right. To see the design and crafting of a software, program or function as a creative act in which the medium is code. This line of thinking has a long history dating back at least as far as the 1970's with Alan Kay's *Smalltalk* programming language. In the last twenty years, it has been centered in the Creative Coding movement, perhaps starting with John Maeda's development of the *Design by Numbers* software/language at MIT's Aesthetics and Computation group in 1996. Other examples include: *Processing* developed by Casey Reas and Benjanmin Fry, *openFrameworks* developed by Zach Lieberman, and P5.js developed by Lauren Lee McCarthy. These platforms are free and open-source. Instead of designing software as a top-down product to be consumed, they design software in which development and extension takes place in the community. Furthermore, all of these examples were developed as a means of allowing artists and designers to access code as creative medium. In the same way a writer learns the syntax and semantics of their given language while simultaneously learning to infuse the words with creative thinking and intent, a designer or artist working with code learns the technical aspects of writing code while simultaneously learning to think imaginatively through code. This semester we will test this paradigm by working with code as our medium. We will design programs that make drawings, objects, parts of buildings, images and perhaps forms of representation that we have not yet anticipated.

Not Magic:

The tool argument becomes even more problematic in a digital context, in which so much of the design and operation of off-the-shelf software remains opaque to the user. The design of the user interface, the naming of commands, and the logic and biases inherent the programming of the software are inaccessible to most users, but remain key components in the design process. Consider for instance the "Boolean" command family in Rhinoceros 3D. It is one of the most commonly used commands in digital modeling. It allows the user to add, subtract, or union a set of solid objects. McNeel, the company that developed Rhinoceros, is careful to warn that the Boolean command is "not magic". However, the use of the word "magic" itself belies a situation in which the user/designer executes a command in blind faith and is either satisfied or not with the results. The command is a combination of a set of other commands: Intersect, Split, Delete, and Join. The commands have been strung together to produce a particular result in which the user is provided a limited set of possible options. These four commands have also been built-up from other lower level sets of commands and geometric operations in a continual process of simplifying to make the command user friendly. This is the process of encapsulation, in which the complexity of programming is encapsulated within a black box, allowing the user a clear and limited scope of interaction. To express this in terms of quantities, the McNeel wiki page that describes the tool in plain English is 2,265 words. All the words that appear in the command prompt when the user calls the Boolean command add up to seventeen. This process of simplification is a process of design. The seventeen words that the user is provided and the options that they are allowed are carefully selected to provide a digital product that is at once easy to use and easy to grasp. It is not the design of a tool, but rather the design of a medium that allows the user to simulate physically sculpting an object with the added benefit of being able to see the object in both rendered and line-based views simultaneously. The design of the medium has as much to do with the design of the object as does the intent of the user.

Computation Media

Computational media uses these traditional human media simply as building blocks to create previously unimaginable representational and information structures, creative and thinking tools, and communication options. (Lev Manovich, *Software Takes Command,* 2010)

This semester we will investigate the process of design through the lens of computational media. As with all degree project research seminars, students will identify a program, site, and a design problem. In addition to this, we will research, develop, and test custom computational media. The computational media will take the form of plug-in that we will add-on to an existing software. The plug-in will be the design project for the fall the semester and visual or physical products will be one test of its efficacy.

In order to develop a critical understanding of software and computational media, students will pursue a line of inquiry into technologies of imaging and fabrication. To build skills in programming and media students will participate in weekly workshops in which the principles of programming and creative coding will be taught. Fundamental concepts such as functions, loops, and classes will covered. At the same time, students will use the principles set forth in the class to develop a plug-in for an existing software platform. The work for this portion of the semester, will be embodied in weekly sketch assignments, in which students will produce a new program or modify an existing one. A new sketch will be due every Monday.

Engagement

We are in a time in which social, environmental, and public health issues are at the forefront of our daily existence. To engage this we will again turn to the open-source movement. P5.js, a javaScript based creative-coding platform on the web, is designed for the community by the community. Consider this mission statement from its page:

> p5.js is a JavaScript library for creative coding, with a focus on making coding accessible and inclusive for artists, designers, educators, beginners, and anyone else! p5.js is free and open-source because we believe software, and the tools to learn it, should be accessible to everyone. (https://p5js.org/)

The developers of p5.js built the platform to provide access to coding and to allow it to be modified and extended by others. Because it is a web based platform, a user can access it from a public computer such as a library and save their work. It is designed to remove barriers to education and provides community support for users. Similar practices can be found in Architecture with the free sharing of plug-ins and components on sites such as https://www.food4rhino.com/en . In all of these instances, the design of the software is the design project. The intent of the software is not to produce a singular project for a specific user, but instead to design a software capable of producing innumerable solutions for a wide range of users. Similarly, as we develop our projects we will consider the design of the our plug-ins as design projects that uses code as medium to engage the social, political, or environmental problems facing architecture as a discipline.

As we develop our plug-ins we will consider some fundamental questions

1. What architectural problems does the plug-in address?
2. Who is the software for?
3. How will it be accessed?
4. What is the mission statement of the plug-in?
5. What other software exists that is similar?
6. What is the name of the plug-in?

STRUCTURE:

The semester will be broken into four graded components:

SKETCHES:

Borrowing a term from platform *Processing*, students will be required to produce weekly "sketches" in the form of programs that produce images or objects. Sketches are due on the Monday of each class and must include (1) image of the output in .png format and the code file. Students will receive credit or no credit on each sketch. The semester will start with single

paragraph "briefs" for the sketches. However, as the semester progresses, the sketches will consist entirely of updates to the students plug-in. As the term suggests, a sketch is a quick test of an idea. Students are encouraged to think imaginatively and to produce work that does not mirror the in-class examples. The sketches must run to receive credit. Sketches must be submitted to the class one-drive/github repository by 10:30 am of every wednesday.

Assignment 1: RESEARCH

In the first assignment, students will complete a series of weekly computational imaging exercises closely connected to the in-class tutorials. The exercises will have little direct relationship to the students research, but will help to introduce them to the fundamentals of programming. In parallel to this students will identify a social or environmental problem and connect to a problem of architecture. This will be supported by a catalog of precedents that embody the particular problem. Students are encouraged to find precedents both within and without the discipline of architecture.

Formats: .jpg, .py, and .pdf

Assignment 2: BUILD

Students will begin to work on their plug-ins. A series of 3D programming tutorial will help students to deal with issues of thickness, color, and visualization. Students will produce a short video that documents their research, plugin, and project statement. The video will also include documentation of user testing, in the form of images and animations produced by other students outside of the class.

Assignment 3: DEMONSTRATE

During this phase students will test their programs by distributing them as plug-ins to other students in the class. Students will collect the results of the test and modify the program accordingly. Lastly, students will produce a short five minute video that explains their degree project research and software plug-in. It should include images from the test as well as a demonstration of the plug-in in action. Lastly, a the video should conclude with a schedule of work to be completed over the spring semester.

Schedule and Deliverables

Week 1 – August 29 + August 31

Course Introduction:

- What is computation?
- What is computational media?
- What is an architectural problem?
- Introduction to core programming concepts
- Basic syntax in python.
- Reading: John Maeda, *How to Speak Machine, 2019* , Chapters 1-3
- Precedent Research
- Watch Film: _Coded Bias,_Shalini Kantayya, 2020

Week 2 –September 5 + September 7

Variables and Input

- Labor Day September 5//No Class
- Variables
- For loops

- Reading: John Maeda, *How to Speak Machine, 2019* , Chapters 3-6
- Precedent Research/Problem research

Week 3 – September 12+ September 14

Conditionals and Output:

- If, Else, Else if, And, Or
- Boolean Variables
- Export Image: Jpeg, PNG, TIFF
- While Loops
- Color: RGB Cube, Linear Gradients
- Export Image: Jpeg, PNG, TIFF
- Reading: Kat Holmes, Mismatch: *How Inclusion Shapes Design, 2018* , Chapters 1-3
- Precedent Research/Problem research

Week 4 – September 19 + September 21

Other Outputs:

- Export Image Sequence: PNG
- Rendering Modes: Default, P2D
- Compile Mp4: moviepy
- Export vector: pdf/dxf
- Reading: Kat Holmes, Mismatch: *How Inclusion Shapes Design, 2018* , Chapters 4-6
- Precedent Research

Week 5 – September 19 + September 21

Review 1: Precedents, Sketches, and Architectural Problem

Storing and Sorting Information

- Lists: Index, Length
- List operations: Reverse, Insert, Pop, Max…
- Globals
- Reading: Kat Holmes, Mismatch: *How Inclusion Shapes Design, 2018* , Chapters 7-9
- Establishing Mission Statement for plug-in

Week 6 – September 26+ September 28

Classes:

- Objects
- Instances
- Reading: Molly Wright Steenson, *Architectural Intelligence*, 2017, Chapter 1
- Researching Geometry, mathematics, and user interface design of plug-in
- Program Research

Week 7 – October 10 + October 12

Testing and Evaluating

- Github
- Versioning
- Reading: Lev Manovich, Software Takes Command, 2013, Chapter 1
- Researching Geometry, mathematics, and user interface design of plug-in
- Reading: Student Directed based on Research Interest
- Program Research

Week 8 – October 17+ October 19

Building the Plug-in:

- Reformatting program to run as a command in Rhino
- Program Research
- Reading: Student Directed based on Research Interest

Week 9 – October 24 + October 26

Review 2: Plug-in Mission Statement, Architectural Problem, Present Working Plug-in, Building Program

Responding to Feedback

- Diagram of possible routes forward
- List of Changes to be made with schedule for completion.
- Reading: Student Directed based on Research Interest
- Site Research

Week 10 – October 31 + November 2

Working Sessions

- Refine/Develop/Test problems with the plug-in in class
- Reading: Student Directed based on Research Interest
- Video Editing Tutorial: How to produce the demo.
- Site Research

Week 11 – November7 + November 9

Working Sessions

- Refine/Develop/Test problems with the plug-in in class
- Reading: Student Directed based on Research Interest
- Develop Statement
- Site Research
- Distribute Plug-in to test group

Week 12 – November 14 + November 16

Working Sessions

- Refine/Develop/Test problems with the plug-in in class
- Reading: Student Directed based on Research Interest
- Develop Statement
- Site Research

- Collect, Evaluate and Curate results from tests

Week 13 – November 21 + November 23

Working Sessions

- Refine/Develop/Test problems with the plug-in in class
- Produce Demo Video

Week 14 –November 28 + November 30

Working Sessions

- Refine/Develop/Test problems with the plug-in in class
- Produce Demo Video

Week 15 – December 5 + December 7

- Studio Finals//No Class

Week 16 – December 14

- Final Review TBD

FINAL GRADE CALCULATION:

Sketches: 10%

Assignment 1: 20%

Assignment 2: 30%

Assignment 3: 40%

Each Assignment will graded using a rubric composed of the course learning outcomes. A sample has been appended to this syllabus.

Activities:

Lecture:

During lecture the instructor will present a set of concepts illustrated through images and text. Students are expected to participate by taking notes and asking questions. Lectures will be recorded and made available on the course moodle page.

Workshop:

The instrumental focus of the class will require a significant amount of time to be dedicated to learning software. During workshops the instructor will demonstrate techniques related to the design problem. Following the demonstration students will repeat the work of the demo and expand it to address the specific problems of their individual project. Workshops will be recorded and made available on the course moodle page.

Reading Discussion:

Short reading assignments will be embedded in the three assignments. Each reading will help to present fundamentals of design as well as specific issues related to the design problem. In the first class following the distribution of the assignment,

students will be expected to participate in discussion of the reading through the production of a list of written questions. These questions will be used to generate topics of discussion.

Review:

Due to the collaborative nature of the class, reviews will be structured to discuss the work of the collective. They will operate as means to exchange design ideas, discuss the artifacts of design as well as index the terminus a phase of work. Students are expected to attend, discuss and question the work.

Readings

Some of the readings of the semester will be provided as pdfs. Please purchase Kindle versions of the following books:

Kindle Purchases:

Maeda, John. 2019. *How to Speak Machine ___ : Computational Thinking for the Rest of Us*. [New York]: Portfolio/Penguin.

Holmes, Kat. 2020. *Mismatch ___ : How Inclusion Shapes Design*. First MIT Press paperback edition. Simplicity. Cambridge: The MIT Press.

McIlwain, Charlton D. 1971-. 2020. *Black Software ___ : The Internet and Racial Justice, from the AfroNet to Black Lives Matter*. New York, NY: Oxford University Press.

Bibliography:

Not all of the books on this list will be required reading. This is instead a rough a list of references that the students should use to develop their knowledge and research independently.

Bruegmann, Robert. 1989. "The Pencil and the Electronic Sketchboard : Architectural Representation and the Computer." In *Architecture and Its Image ___ : Four Centuries of Architectural Representation ___ : Works from the Collection of the Canadian Centre for Architecture*, edited by Eve Blau and Edward Kaufman. Cambridge: MIT Press.

Cache, Bernard. 2013. "Durer-Vitruvius-Plato. Instruments of Thought." Presented at the Architecture in the Making, KTH School of Architecture, Sweden, April. https://www.youtube.com/watch?v=VxzdW4H4aww.

Canizares, Galo. 2019. *Digital Fabrications ___ : Designer Stories for a Software-Based Planet*. [Novato]: ORO Editions/Applied Research & Design.

Carpo, Mario. 2008. "Alberti's Media Lab." In *Persepctive, Projections, & Design: Technologies of Architectural Representation*, edited by Lemerle, Frederique and Mario Carpo. New York: Routledge.

Casey Reas, and Chandler McWillaims. 2010. *Form + Code*. New York, N.Y.: Princeton Architectural Press.

Garcia-Cuevas, Diego, and Gianluca Pugliese. 2020. *Advanced 3D Printing with Grasshopper: Clay and FDM*. Independently Published.

Hilbert, D., and S. Cohn-Vossen. 1952. *Geometry and The Imagination*. New York: Chelsea.

Holmes, Kat. 2020. *Mismatch ___ : How Inclusion Shapes Design*. First MIT Press paperback edition. Simplicity. Cambridge: The MIT Press.

Issa, Rajaa. 2019. *Essential Mathematics for Computational Design*. 4th ed. Robert McNeel & Associates. https://developer.rhino3d.com/guides/general/essential-mathematics/.

Legendre, George. 2002. *IJP: The Book of Surfaces*. London: AA.

Lostritto, Carl. 2019. *Computational Drawing*. Applied Research and Design.

Maeda, John. 2004. *Creative Code*. New York, N.Y.: Thames & Hudson.

Maeda, John. 2019. *How to Speak Machine ___ : Computational Thinking for the Rest of Us*. [New York]: Portfolio/Penguin.

McIlwain, Charlton D. 1971-. 2020. *Black Software ___ : The Internet and Racial Justice, from the AfroNet to Black Lives Matter*. New York, NY: Oxford University Press.

Menges, Achim, and Sean Ahlquist. 2011. *Computation Design Thinking*. Chichester: Wiley.

Peters, Brady, and Daniel Davis. 2013. "Design Ecosystems: Customising the Architectural Design Environment with Software Plug-Ins." Edited by Xavier De Kestelier and Brady Peters. *Architectural Design*, Building of Algorithinc Thought, 83 (2).

"Rhino.Python Guides." n.d. Accessed August 20, 2021. https://developer.rhino3d.com/guides/rhinopython/.

"RhinoPythonPrimerRev3.Pdf." n.d. Accessed August 20, 2021. https://s3.amazonaws.com/mcneel/misc/docs/en/RhinoPythonPrimerRev3.pdf.

Steenson, Molly Wright. 2017. *Architectural Intelligence ___ : How Designers and Architects Created the Digital Landscape*. Cambridge, Massachusetts: The MIT Press.

Witt, Andrew. 2010. "Machine Epistemology in Architecture." Translated by Annette Wiethuchter. *Candide-Journal for Architectural Knowledge* 12 (3): 37–88.

Woodbury, Robert. 2010. *Elements of Parametric Design*. Routledge.

Zelle, John M. 2017. *Python Programming ___ : An Introduction to Computer Science*. Third edition. Portland, Oregon: Franklin, Beedle & Associates Inc.

Estimate of Costs

The cost of the course will be:

Purchase (3) Kindle Books = $35

3D printing = $100

Netflix subscription for 1 moth = $10

Total = $145

Course Policies:

Attendance:

As a class focused on the acquisition of instrumental knowledge, attendance is central. Students will be graded on attendance during each of the three phases of work. Each missed class that has not be an excused will result in a 1 grade point deduction in the category of Professionalism. More than 2 absences during a given phase of work will result in a "0" for Professionalism.

Submitting Late Work:

All work must be digitally submitted to moodle on the date provide on the assignment. One-half letter grade will be deducted for each 24 hour period following the submission deadline. Work later than 72 hours will receive an "F."

Use of cell phones, or other technology during the class meeting:

The use of cell-phones, tablets, or computers is permitted as long as it pertains to the work of the class.

Formatting or quality of submitted work:

All work must be submitted digitally. Each assignment should be posted to Sharepoint prior to the first-class meeting directly following each review. Students may make changes to the work after the review to improve their grade based on comments received at the review. No changes will be accepted after the due date. Please see assignment description for file format for each assignment.

DEPARTMENT POLICIES AND PROCEDURES

Requirements for Documentation and Archiving

Each student must submit documentation of the full semester's work at the end of each term, in pdf format. Materials should include research, writing, and design work, including important study models and sketches. Studio faculty will further define how this work should be organized and presented before the end of the semester. Failure to submit the required documentation in usable format may result in a grade reduction in the final grade of the semester. Documentation of the studio work is essential for the NAAB accreditation process and assessment of the architecture program.

The university reserves the right to retain student work for archival purposes. Projects/models, assignments, and exams will be kept at the department's discretion for this purpose. Students will be asked to help with archiving their projects at the end of the semester.

Writing Requirements

All written work must meet the standards for English. Poorly written papers may be returned without a grade for revision and resubmission, and may be subject to grade reduction. Students are encouraged to utilize the Woodbury Writing Center. Link is here.

Studio Culture

The studio environment is an essential component in learning to become an architect. One goal of the School of Architecture is to create a vibrant, exploratory, safe and respectful learning culture for students. Only through respect between faculty and students, as well as students among themselves, can a healthy educational studio culture be fostered. Students are required to uphold high standards of behavior and academic discipline while in the studio. See the full Studio Guidelines and Studio Culture Policy for more information.

School Policy on Social Equity and Diversity

Our mission is to provide an environment where people can learn, teach and work with a shared sense of purpose, core values and respect without bias towards individual beliefs, values and areas of difference. We do this in an effort to create a community that respects and values the full and equal inclusion of its members. Our goal is to provide an environment that is welcoming and inclusive of all.

Universal Pedagogy

Woodbury University is committed to making reasonable accommodations to assist individuals with disabilities in reaching their academic potential. Students desiring accommodations due to a physical, learning or psychological disability must first

complete an Accommodations Request Form, which can be downloaded from http://go.woodbury.edu, and found under "Academic Resources." Accommodations cannot be granted prior to the instructor's receipt of a Notification of Academic Accommodation Plan (NAAP) from the Disabilities Coordinator. Accommodations are never provided retroactively. (For more information, contact the Disabilities Coordinator in the Whitten Center (818) 394-3345.)

Accommodations Policy

Woodbury University is committed to making reasonable accommodations to assist students with disabilities in reaching their full academic potential. To that end, your instructor will comply with official requests for accommodations. Please follow the instructions found on thisAccommodations and Documentation pageor https://woodburyedu.sharepoint.com/sites/ODAS/SitePages/Accommodations.aspx so that your instructor will be provided the appropriate instructions for your accommodations.

Academic Honesty

Academic integrity is important for two reasons: first, independent and original scholarship ensures that students and scholars derive the most from their educational experience and the pursuit of knowledge. Second, academic dishonesty violates the most fundamental values of a community of scholars and depreciates the achievements of the entire University community. Accordingly, Woodbury University views academic dishonesty as one of the most serious offenses that a member of our community can commit. Adherence to the Academic Honesty Policy reflects the commitment of our community to the ideals of learning, research, and scholarship. See Catalog for the entire Academic Honesty Policy.

Grade Requirements

Refer to the Woodbury University catalog for grading standards and policies.

Environmental Responsibility

Studio projects shall be designed in a socially and environmentally responsible manner. All projects should reduce dependencies on non-renewable resources.

Class Attendance

It is mandatory that students take advantage of all scheduled course time. Regular attendance at EVERY class is expected throughout the duration of the class/studio time. Arrival at the beginning of the class period is required. Lateness or early departure may be considered as an absence. 'Regular and prompt attendance at all university classes is required. The instructor is not obligated to assign extra work or to prepare additional examinations for classes missed. It is understood that when 15% of the class time has been missed, the student's absence rate is excessive. Each instructor will announce his/her attendance policy in the course syllabus.

Excused Absence

Students should report any illness or emergency to their course instructor, preferably before missing the class, by emailing the instructor. Medical reasons for absences can alternatively be shared with the school nurse and/or the coordinator for the ODAS (disabilities office) who will notify instructor of receipt without revealing specific information.

Students who anticipate absence due to religious observance or similar commitments should speak with their instructor at the start of the term to review all dates in question and develop a plan to meet all course requirements.

Email

Students are advised to meet with their instructors during posted office hours. Face-to-face communication in discussing and resolving problems is preferable to email exchanges. Additionally, meetings must be scheduled in advance using email

correspondence. Email correspondence must be written in a respectful and professional manner. It is the student's responsibility to consistently check for email. students should use their official woodbury.edu email account.

Due to confidentiality and FERPA requirements all faculty, staff and students, when corresponding through email, must use their university provided Woodbury.edu email accounts. Students are encouraged to check this email address regularly as it is the only email address in which they will receive official course or university information.

Grievance Protocol

Students should use the following protocol for questions, grievances, or general concerns about coursework and the studio environment. Health and safety concerns and emergencies should immediately be directed to campus security (818-252-5208). Academic concerns should be directed first to the student's instructor, and then to the studio coordinator as appropriate. If further consultation is required, the student is advised to meet with the Coordinator and/or the Chair.

Class Syllabus and Structure

While every effort will be made to follow the outline of the published syllabus, course structure and calendar may be changed at the instructor or coordinator's discretion. Announcements will be made if such changes occur. Students who miss class are responsible for tracking any such announcements.

ONLINE COURSE DELIVERY POLICIES AND GUIDELINES

Formats of delivery:

Synchronous online: All instruction is provided via the Internet and no face-to-face instruction is required. Faculty and learners meet for regularly scheduled class sessions.

Synchronous activities provide real-time dialogue that can provide the human interaction that is needed among our Woodbury student population.

asynchronous online : All instruction is provided via the Internet and no face-to-face instruction is required. Faculty and learners do not meet for regularly scheduled class sessions.

Asynchronous activities allow students with the flexibility to complete certain course work at their own pace, within reason. It also allows students time to think, write, and reflect.

hybrid with synchronous online: An instructional delivery method which combines the traditional delivery and the synchronousdistance delivery formats.

hybrid with asynchronous online: An instructional delivery method which combines the traditional delivery and the synchronousdistance delivery formats.

traditional: This delivery method allows learners and faculty to meet in person or as a group for regularly scheduled class sessions either on campus or at another physical location.

Type of course:

Lecture : Refers to the first or primary organization of non-lab class instruction, e.g., a lecture where instructor-based material is presented, or a seminar where material is analyzed and discussed by both students and instructor. Also includes case studies and team-based learning situations. Class meeting time equals 50 minutes per unit per week.

Studio : Refers to situations where the student is engaged in the practice and use of techniques for productions in the areas of architecture, interior design, graphic design, and other design forms. This instruction is used to further advance student's

skills in their field of design. The instructor role varies from direct assistance to simple availability for questions and supervision. Class meeting time equals 100 minutes per unit per week.

Laboratory : Refers to the first organization of laboratory class instruction unless one of the other classifications above is more appropriate. Includes both group instruction and individualized instruction such as biology and physics lessons, supervised computing exercises, and hands-on activities. Class meeting time equals 50 minutes per unit per week.

Length of sessions:

16-week (Fall/Spring)

7-week intensive format (Fall/Spring)

5-week intensive format (Fall/Spring)

10-week (Summer)

6-week (Summer Super Sessions)

Online / Hybrid Requirements

Students must have basic computer skills, including the use of word processing software, email, and the ability to use internet browsers, such as Safari, Firefox, or Chrome.

All communication by email will be using the assigned woodbury.edu address. Students are required to access this email account on a daily base to ensure timely communication.

Woodbury University's Learning Management System (LMS) of record is Moodle. Moodle should be used to provide students information they need to plan, prepare, and learn in the course. This information includes, but is not limited to: (a) your course syllabus; (b) assignment due dates, instructions, and grading rubrics; (c) course schedule; (d) additional course materials and links, etc.; and/or (e) exams.

Students are responsible for meeting the technical requirements of Moodle and Microsoft Teams and to familiarize themselves with the Moodle Learning Management System and Microsoft Teams Communications System.

IT provides a Moodle (and Microsoft Teams for online delivery) orientation "course" visible to students enrolled in all courses. Students unfamiliar with Moodle are required to review or consult it as needed. This ensures class time is

dedicated to course content and not technical tutorials. The syllabus should also indicate how IT will support

students' online technical needs.

All required materials, including readings, videos, lectures will be posted on and can be accessed through Moodle.

All assignments have to be submitted through Moodle or OneDrive, depending on file size and faculty instructions.

Exams and quizzes will be administered through either Moodle or Proctorio. Students are required to install the Proctorio Extension in advance.

Attendance policy

Regular and prompt attendance at all University classes is required. It is the responsibility of the student to adhere to class/studio participation expectations. The instructor is not obligated to assign extra work or to prepare additional content

for material missed. It is understood that when 15% of the class time has been missed, the student's absence rate is excessive. The interpretation of this participation policy is left to the discretion of individual faculty.

Attendance of synchronous classes will be measured through attendance of the Microsoft Teams meetings. Students are required to have their camera turned on for the entire class session to be counted as present. If technical or privacy issues prevent the student from having the camera turned on, then the student must contact the Office of Student Affairs and apply for an exemption. Students must complete an Accommodations Request Form, which can be downloaded from http://go.woodbury.edu, and found under "Academic Resources." Accommodations cannot be granted prior to the instructor's receipt of a Notification of Special Needs Release Form from the Disabilities Coordinator. Accommodations are never provided retroactively.

Attendance of asynchronous classes will be measured through the last access to the course in Moodle, participation in quizzes and online discussions, as well as submitted assignments by the required day and time.

Protecting Privacy and Data During Remote Instruction

This class is being conducted over Microsoft Teams and Moodle. As the host, the instructor may be recording the sessions. The recording feature for others is disabled so that no one else will be able to record the sessions. No recording by other means is permitted. The sessions will be posted on the Moodle class website unless otherwise notified. In case of privacy concerns and individual students wanting not to appear in the recording, the student must contact the Office of Student Affairs and apply for an exemption. Students must complete an Accommodations Request Form, which can be downloaded from http://go.woodbury.edu, and found under "Academic Resources." Accommodations cannot be granted prior to the instructor's receipt of a Notification of Special Needs Release Form from the Disabilities Coordinator. Accommodations are never provided retroactively. If the student prefers to use a pseudonym instead of the real name, please let the instructor know what name will be used so that the instructor knows who you the student is during the session.

Pursuant to the terms of the agreement between the vendors (Moodle and Microsoft Teams) and Woodbury University, the data is used solely for this purpose and the vendor is prohibited from re-disclosing this information. Woodbury University also does not use the data for any other purpose. Recordings will be deleted when no longer necessary. However, the recording may become part of an administrative disciplinary record if misconduct occurs during a video conference.

Calculation Of Grade

Letter grades are converted to numeric values using the following values:

Letter	GPA	%	Definition
A	4.00-3.84	96-100	Student learning and accomplishment far exceeds published objectives for the course/test/assignment and student work is distinguished consistently by is high level of competency and/or innovation.
A-	3.83-3.50	92-95	
B+	3.49-3.17	88-91	Student learning and accomplishment goes beyond what is expected in the published objectives for the course/test/assignment and student work is frequently characterized by its special depth of understanding, development, and/or innovative experimentation.
B	3.16-2.84	84-87	

Letter	GPA	%	Definition
B-	2.83-2.50	80-83	Students learning and accomplishment meets all published objectives for the course/test/assignment and the student work demonstrates the expected level of understanding, and application of concepts introduced.
C+	2.49-2.17	76-79	
C	2.16-1.84	72-75	
C-	1.83-1.50	68-71	Student learning and accomplishment based on the published objectives for the course/test/assignment were met with minimum passing achievement.
D+	1.49-1.17	64-67	
D	1.16-0.60	60-63	
F	0.00-0.60	< 60	Student learning and accomplishment based on the published objectives for the course/test/assignment were not sufficiently addressed nor met.