GUIDED PROJECT TRAINING GUIDE FOR INSTRUCTORS

Updated June 2020

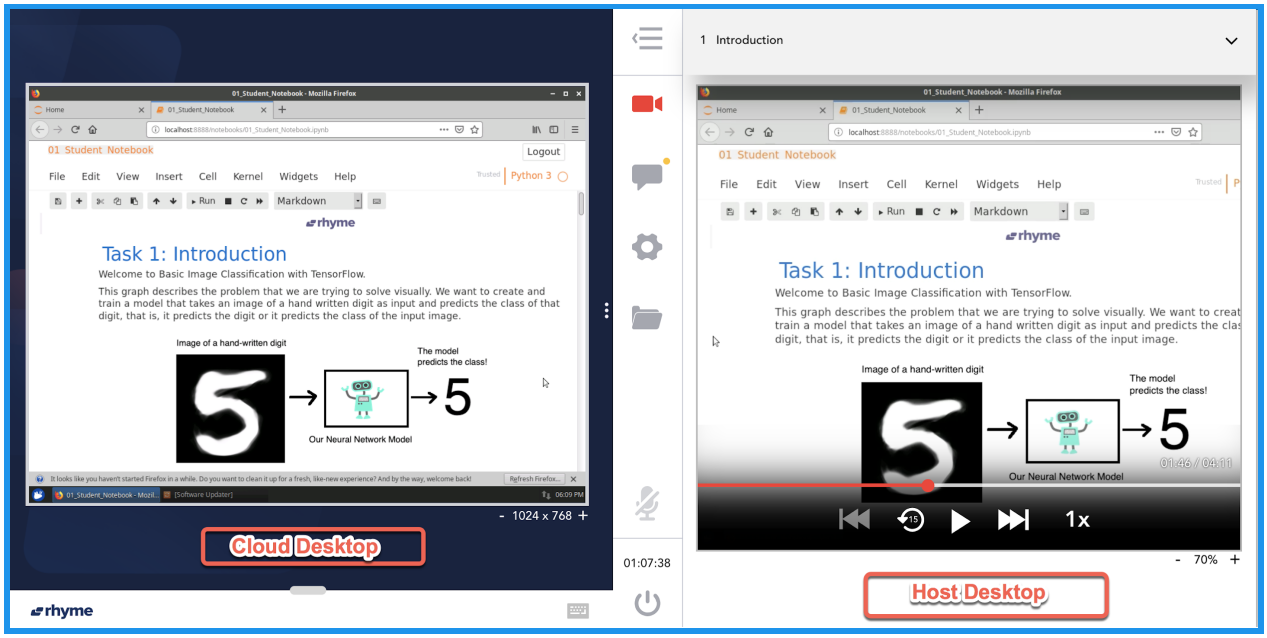
Contents

Guide Overview

This guide was written to help Rhyme instructors understand the Rhyme platform, how to use Rhyme, and best practices for creating Guided Projects.

What is Rhyme?

Rhyme is an online platform for hands-on projects. With Rhyme's virtual machines, beginner to intermediate-level learners can follow along with self-paced or live guided sessions while simultaneously completing a project or assignment — all from one browser using pre-configured Windows or Linux cloud desktops. Rhyme truly embraces the concept of active learning or "learn by doing."



Rhyme Student View

When to Use Rhyme

Rhyme is a great teaching option in following cases:

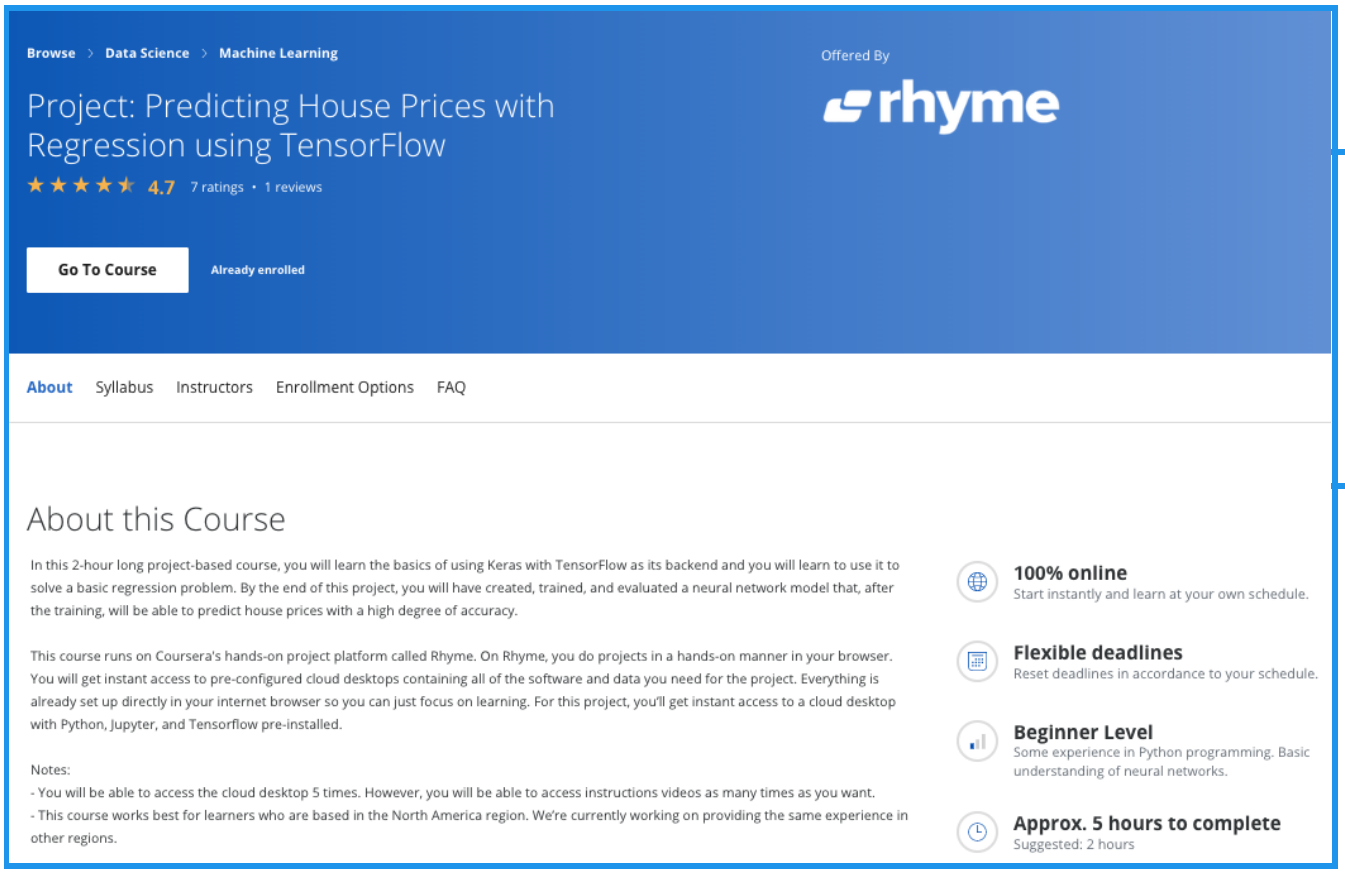
- **Learners must learn the skills through hands-on activity instead of solely listening and watching.** For example, you want to teach learners how to create a pie chart in Tableau. The best way to do so is giving learners a data set, letting them play with Tableau's interface and creating a pie chart by themselves.

- **Learners need to focus on the knowledge itself instead of setting up the working/coding environment.** For example, you are introducing python to learners who are new to coding. You don't want learners to lose interest by installing the IDLE. You can use Rhyme to let learners jump into coding directly and have small win by printing "Hello World"

- **Learners benefit most from observing and following an expert.** Experts structure knowledge and approach problems in different ways from novices. The best way for learners to adopt experts' mindset is having them observe and follow along. For example, debugging could be painful for new programmers. Watching how a seasoned engineer debug the code systematically and repeating the process on their own could be super helpful for learners.

Creating a Rhyme course on Coursera

- **You will create a Rhyme project and launch it in a course shell on Coursera.** The Rhyme project will be developed on the Rhyme website and linked to the Coursera platform via an LTI in a course shell.



- You will want to design your project first before recording it in Rhyme or writing any code. Use this **Project Script Template** to draft your project.

- Your Coursera Project Manager will help you set up your course shell on the Coursera platform.

- Once you create your Rhyme project, you will also create a graded quiz to test the topics introduced in the project. Use this **Project Quiz Template** to create your quizzes or do so directly on the Coursera platform.

- The course will also include learning objectives, readings to help guide the learner, and more information about how to be successful in the project. Use the **Project Launch Checklist** to help you prepare the launch of your project.

Tools	Learner Experience	Grading	App Type	Use Case	Infrastructure	Connectivity
Rhyme	Follow-along video & learner workspace	Graded* & Ungraded	Desktop, VM	Industry-Standard Software	VM, Virtual Browser	Open network access
Course shell	All Coursera content outside of Rhyme, e.g. readings and quizzes	Use a graded quiz to evaluate Rhyme project	Coursera mobile app and via browser	Learners access the project here	Coursera's main authoring interface	Limited network access

Overview of Rhyme and Coursera content

How to Start Authoring a Project with Rhyme

Creating your first Rhyme project will take some time. Content creation time varies depending on what you're teaching but in total, you should expect to spend about 50 hours to create a Rhyme Project from scratch for the very first time. After that, you'll be familiar with the process and Rhyme capabilities and create a project much faster. To help you create your project, we've outlined the steps you should take for content creation.

Step 1: Create a course outline (6-8 hours)
Just like you need a blueprint to build a house, a project outline is essential for you to create a coherent and engaging Rhyme learning experience. A project outline usually includes:

- Learning objectives of the project. i.e. “By the end of the project the learners will be able to...”
- Difficulty level. i.e. Introductory, Intermediate, Advanced
- Prerequisites. i.e. the learners must proficient programming in Python, etc...
- Name and order of each task within the project
- Learning objectives of each task
- A description of the coding examples that will be used (i.e. coding example showing how to find phone numbers using regular expressions).

⭐ ***Tip:*** *Use the first section of the* ***Project Script Template*** *to map out an overview of the project.*

Step 2: Flesh out the content (10-16 hours)
Now you have a list of tasks layed out. It’s time to create all the materials needed for learners to complete each task. It may include preparing the data set, writing the code, searching for relevant resources, etc.

⭐ ***Tip:*** *Start outlining the Storyboard section of the* ***Project Script Template*** *to map out an overview of the project.*

Step 3: Script the tasks (5-8 hours)
Some instructors find it helpful to script the lessons ahead of time, and to record in small chunks (about 30 second snippets) and use the keyboard shortcuts to pause/resume recording.

⭐ ***Tip:*** *Start Storyboard section* ***Project Script Template*** *to map out each Task and make notes about the VM visuals and accompanying quiz questions.*

Step 4: Practice before recording (4-6 hours)
It is recommended to practice your lesson (including your mouse movements) before recording. Remember that the highlighted window on the virtual desktop moves when you move the mouse, so you also have to practice and be conscious of how to move the mouse while you record since this will change what the learner will see in their small window. Therefore, practicing beforehand will minimize the number of retakes and can save you a lot of time.

⭐ ***Tip:*** *Use keyboard shortcuts or hotkeys to help you during the recording session. Using a dual-monitor or printing out the script can also be handy during recording!*

Step 5: Record and create sessions (6-10 hours)
Once you feel ready, record the videos for each task and create sessions for learners. See our Video Editing Documentation.

⭐ ***Tip:*** *Make sure you give learners the opportunity to practice each task; you can tell them to pause the video and try it out before moving on.*

💡Authoring Best Practices

Project Introduction

1. A learner should be able to complete a Rhyme project in less than an hour (60 minutes).

2. Start each project with a vision of the end-goal. Get learners excited about the project by showing a completed demo of what the project will look like at the end or explain why it's worthwhile to complete this project.

Designing Tasks Effectively

3. Keep each task short to give learners more bite-sized content and more control. We recommend each task is 4 to 7 minutes long. Tasks longer than 7 minutes can get tiring for learners to follow along. They also take longer to record and save.

4. Each task must be clearly labeled both in the video and on the cloud desktop. For example, each cell in a jupyter notebook should label, "Task 1: [code]" etc. Introduce the beginning of each video with the learning objective, e.g. "In this task, we will create..."

5. At the beginning of the next task, quickly acknowledge the learner and say, "welcome back; in this task, we will... [clearly state the specific learning objective for the task]."

6. At the end of a task, say a quick sentence on what we'll do in the next task. This will let learners know what to look forward to.

7. Give learners the opportunity to practice each task before you give the solution. Tell the learner to “pause the video” and try it out before moving on to the next step or revealing the answer. Keep in mind, beginner learners will need more step-by-step guidance than an intermediate audience. Be sure to include explanations of “why” learners are doing the steps they are doing; this helps learners see the connection between smaller steps and the project’s overarching goal.

8. Keep the project connected to the real-world. Give examples from the real-world. Ask yourself, how will learners take what they’ve learned here and apply it into their daily life? Again, be sure to explain the importance of what learners are doing along the way; contextualize each specific task within the framework of the larger project. Let learners know how the task contributes to the end goal.

Using the Cloud Desktop

9. Work toward the solution while keeping sight of the end goal. Help the learner see, from time to time, the progress they're making. For example, if you're making a webapp, every few minutes run the webapp to see how it runs or looks now. If you’re writing a programming function, write a short version of it and run it to verify that it’s working. Then, add more behavior in the function progressively.

10. Remember, the learner is following your actions and key presses. Verbally say what you're doing while you're doing it; and move and type keys slowly. After every set of actions, let learners catch-up by saying, “take a moment to practice this task by pausing the video right now.”

11. Do not use keyboard shortcuts that may not be available to learners using Windows or Mac keyboards. For example, if you use a keyboard shortcut that uses the <Windows> key, then learners using Mac keyboards won't be able to press that key while following you. If you do use a keyboard shortcut, verbally state the shortcut when you use it; this cues learners on what to do since they cannot directly see you using the shortcut.

12. Keep in mind which operating system is running on the cloud desktop. For example, if you’re using a Windows cloud desktop, then to copy text, even Mac users will need to press Control-C (not Command-C, as they’d expect). Another reason to not use keyboard shortcuts. Again, if you do use a keyboard shortcut, verbally state the shortcut when you use it; this cues learners on what to do since they cannot directly see you using the shortcut.

13. To start or pause a recording, use the keyboard shortcut Alt-Shift-R. This way, your recordings won't show the cursor coming from the bottom corner at the start of recording and going to the bottom corner at the end of recording.

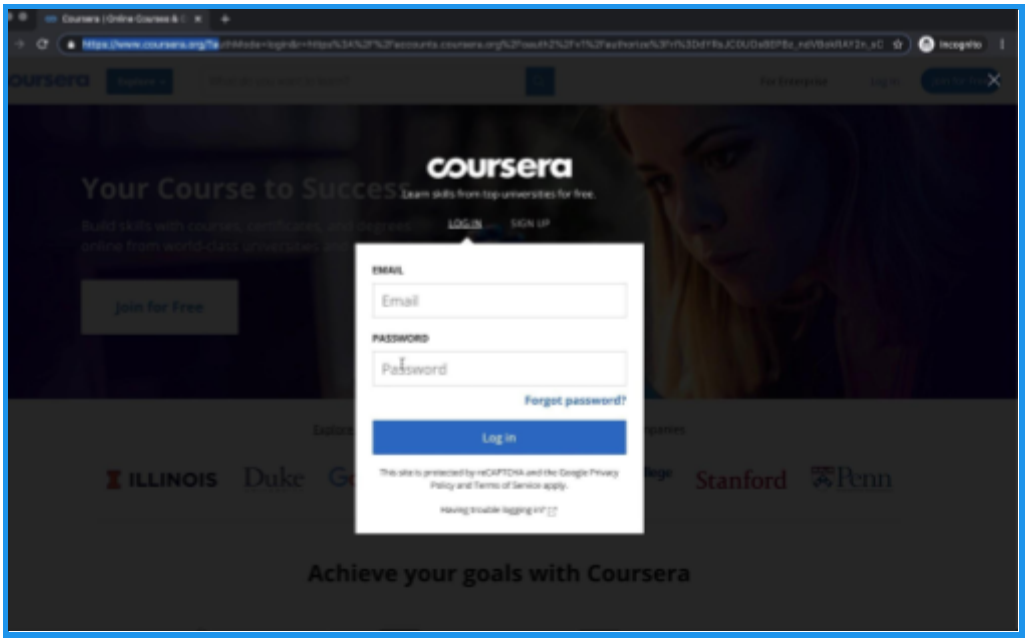
14. Congratulate the learner for a job well done when they finish all the tasks and have completed the project! Provide a quick overview of what they learned and how they will be able to apply that skill into their daily life.

Using the Rhyme Platform

Login into Rhyme

Go to rhyme.com then click "Dashboard". Be sure you are logged into your instructor account on Coursera.

See video below by clicking images



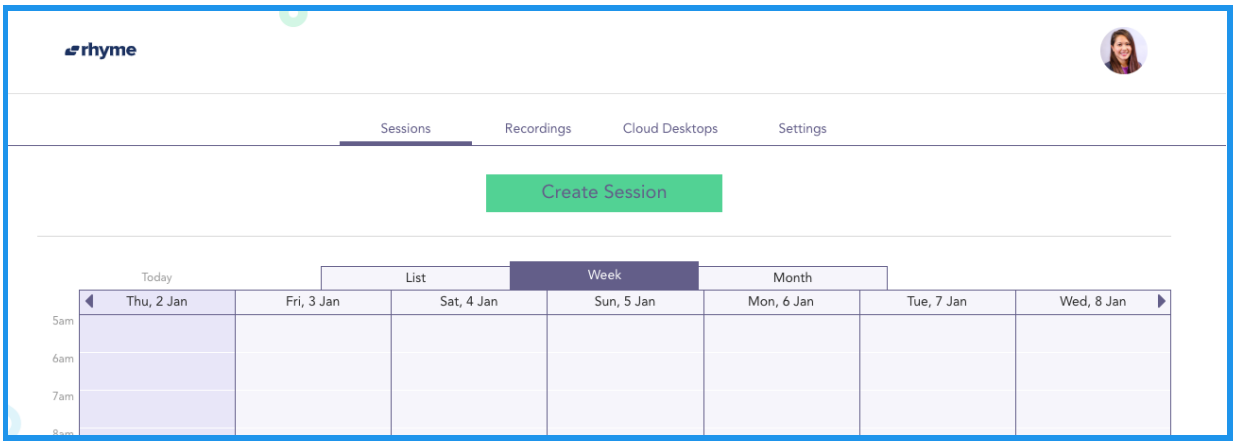
Terminology

Here are some key terms you will encounter in the Rhyme environment.

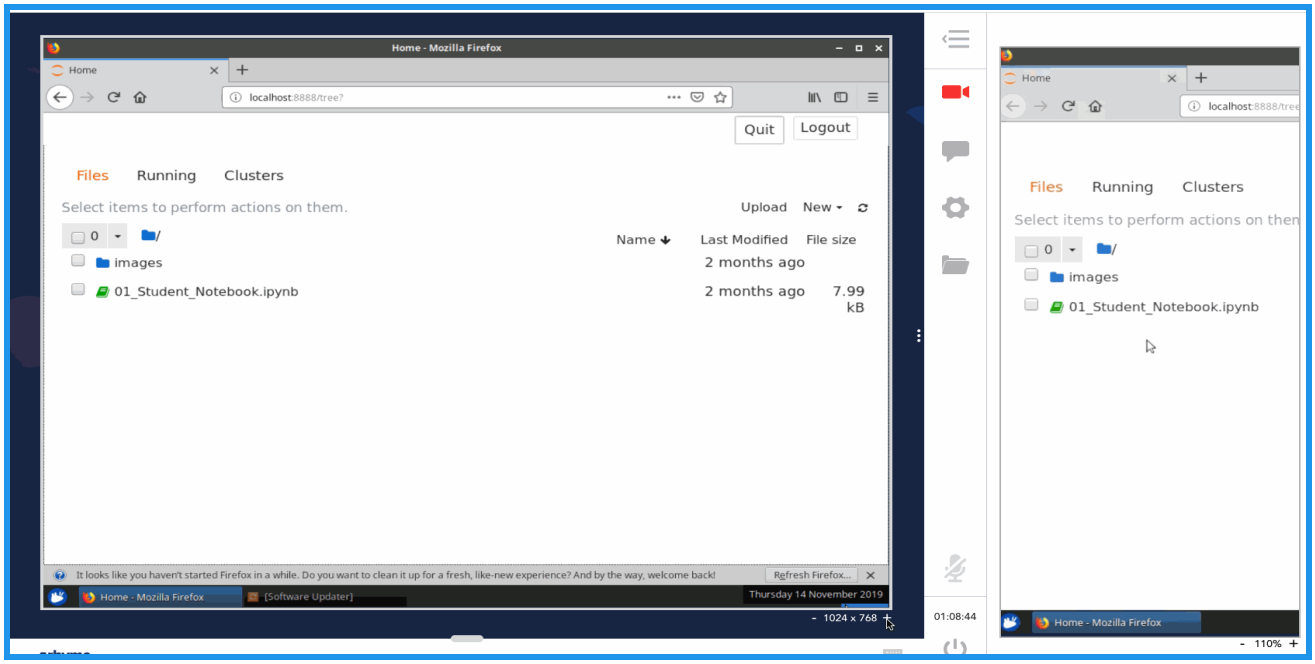
Host- Instructor leading the session sharing their desktop or have recorded their session

Dashboard- Rhyme dashboard where you can create a session and also access the four tabs:

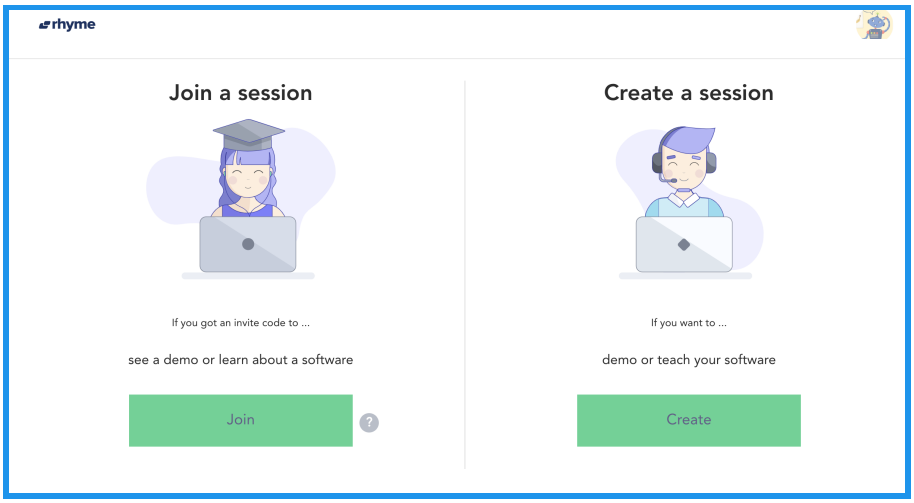
Sessions tab- calendar with scheduled sessions
Recordings tab- all recordings of the user. You will need 1 recording per project.
Cloud Desktops tab- shows a grid with all custom cloud desktops authored by user
Settings tab- a user can change his/her name, photo, credit card, about information, social links, and set a new password



Cloud desktop- The main component of the session app is the cloud desktop Screen. This screen is on the left side and in it the user can see and access his/her Cloud Desktop. Hosts may also view and control learner cloud desktops in live sessions.



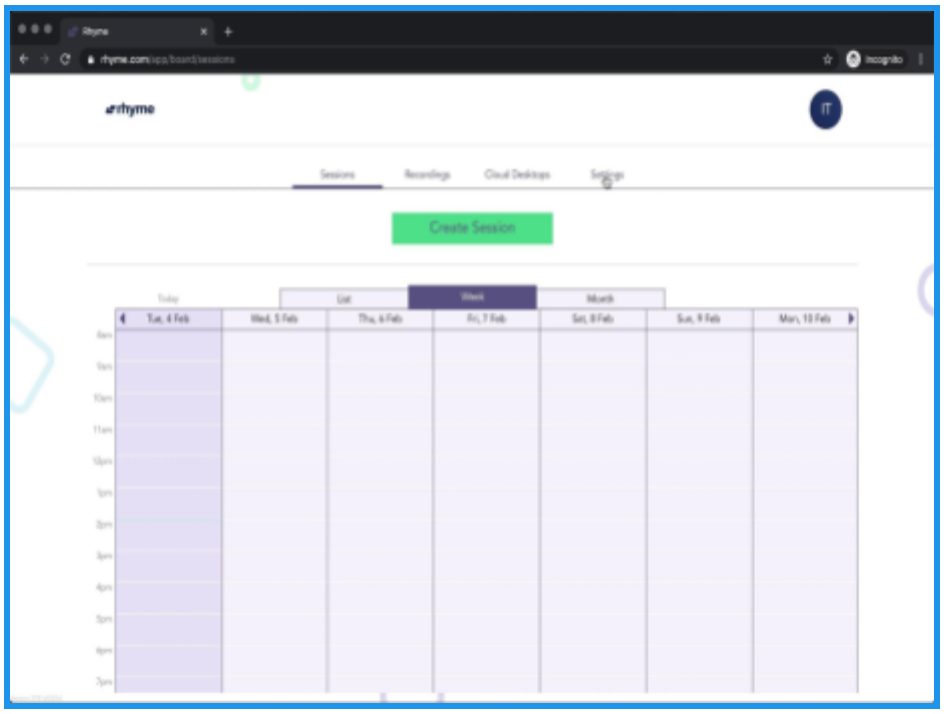
Session- When a user logs in for the first time, they will have an option to either create or join a session. The user can choose "**Join**" where they will be prompted to enter the secret of the session they would like to join or they can click on "**Create**" and be redirected to the Author app.



Task- an activity you want the learner to complete. In a Rhyme project, you will need to create one recording item and build 5-7 tasks for learners to complete in that recording. It is recommended that each task is no longer than 7 minutes.

Account Settings
This section is auto-populated, except for the credit card information. If you are an instructor working with Coursera, please reach out to your Coursera Project Manager to obtain a virtual credit card number. This will allow you to create Cloud Desktops.

See video below by clicking image



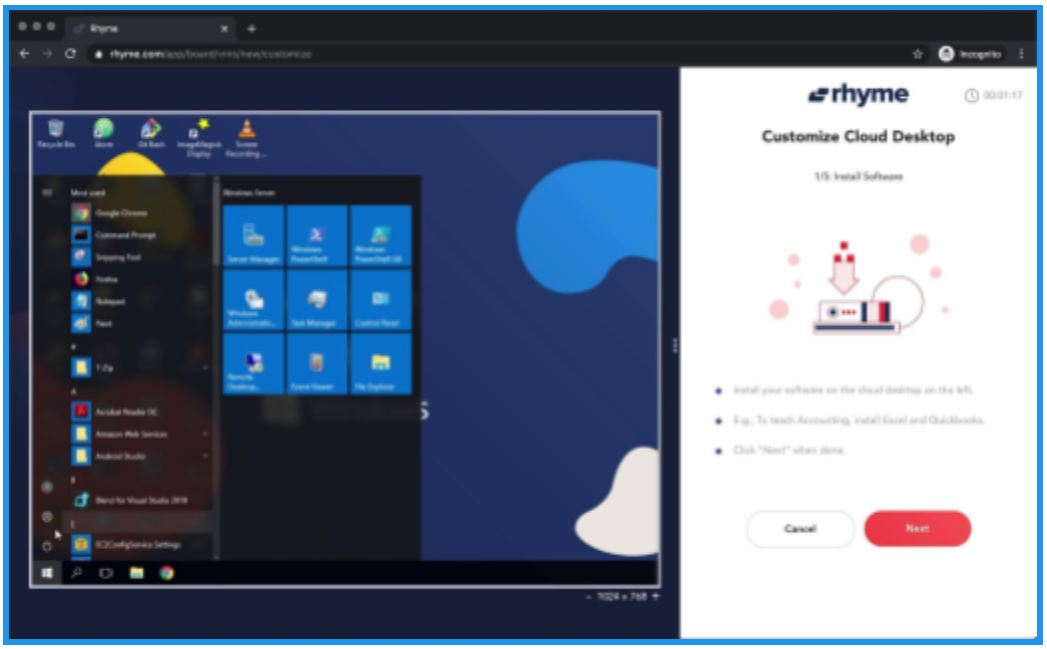
How to Create a Cloud Desktop Configuration in Rhyme

Visit the link below to follow step-by-step instructions on using Rhyme.
https://rhyme.com/how-to-use-rhyme

⚠ **Reminder:** **NEVER** log in to or use your personal accounts on the cloud desktop, both when you record a video and when you customize a cloud desktop. After all, the student will have an identical copy of the cloud desktop that the instructor customized.

Before you begin creating the project, please read the following best practices and check out our video recording tips.

Create a Custom Cloud Desktop



Click image to play video

1. Create a new folder on the desktop to put the files you'll use. Give friendly names to the folder and the files you upload there.
2. If the Rhyme app's file-upload doesn't work, you can upload the files using a cloud-based service like https://uploadfiles.io.
3. Do not log into your email, slack, dropbox, etc. accounts in the cloud desktop. If you do, please remember to logout before saving the cloud desktop.
4. Inside the new folder on desktop, order the files so they are easily accessible to the learners. E.g., create folders per task or number the files.
5. Delete unnecessary files (e.g., installers) and empty Trash before saving cloud desktop.

If using web apps on Chrome (Windows):

1. Create a new shortcut on the desktop with the target "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" --kiosk --incognito https://www.squarespace.com/ . Note: Here, we're assuming a project on Squarespace. Kiosk, so that it is in full screen mode and incognito for security reasons.

2. Test the shortcut to make sure that it in fact opens in kiosk incognito mode to the desired URL.
3. Put new browser shortcut in startup folder (for Windows, this is in "C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup"). Copy the shortcut into this folder (Don't move, i.e., also keep a copy of the shortcut on the desktop).
4. When taking a screenshot of the cloud desktop, launch the Chrome shortcut created earlier so that Chrome and the web-app are visible when the screenshot is taken.

If using web apps on Firefox or Chrome (Ubuntu):

Note: These same steps are applicable to auto-launching a website on Chromium. You just have to replace firefox with chromium-browser --password-store=basic.

1. In the Start Menu, search for startup and you will see Session and Startup. Click on it.
2. Navigate to the Application Autostart tab.
3. Click on Add.
4. Name: Firefox Autostart. You can leave the description blank. Command: firefox https://www.example.com. Replace example.com with the URL of your choice.
5. Click OK followed by Close.

If using installed apps (like Excel):

1. Install the required software and apply for a license that'll work even when multiple cloud desktops use it.
2. Optional (if not going to use taskbar for the project): Right click on the taskbar, click on properties, click the box next to "auto hide the taskbar," uncheck "lock the taskbar." Click apply and then OK.
3. Move the installers to trash.
4. Disable auto-updates for all software you installed. Update notifications will just disrupt your learners. Instead, you can just update the software in the master custom cloud desktop every so often. That way, the update is reflected for all future learners automatically.
5. Ensure that the file types used in this project are opened by default by your installed software. You get this option if you use "Open With" in Finder.
6. Put new shortcut in startup folder (for Windows, this is in "C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup"). Copy the shortcut into this folder (Don't move, i.e., also keep a copy of the shortcut on the desktop).

7. When taking a screenshot of the cloud desktop, open the files in the desired software and make sure it is in an interesting state when screenshot is taken.

If launching Jupyter Lab on boot (Ubuntu):

1. The installation script `installjupyterlab.sh` can be found on the Desktop.
2. To run the script, open the Terminal Emulator and type `bash Desktop/installjupyterlab.sh`
3. The script will install Python, pip, Jupyter Lab, and create a `Project` folder on the Desktop.
4. Place your Jupyter notebooks and project files within this `Project` folder.
5. Save your Cloud Desktop to launch Jupyter Lab on boot.
6. To launch a specific notebook, edit the `/etc/xdg/autostart/jupyterlab.desktop` file and add file path of your notebook after `Exec=jupyter lab`
7. For example, if your notebook is called `notebook.ipynb`, the corresponding entry in `/etc/xdg/autostart/jupyterlab.desktop` is `Exec=jupyter lab /home/rhyme/Desktop/Project/notebook.ipynb --NotebookApp.notebook_dir=/home/rhyme/Desktop/Project/`

Launching Jupyter Notebook (or Lab or any other app) at startup on Windows

1. Create a new shortcut.
2. Assuming the instructor has a folder `C:\Users\Rhyme\Desktop\Notebooks` in which they need to launch Juypyter Notebook, we can enter this command in the shortcut: `C:\Windows\System32\cmd.exe \k cd C:\Users\Rhyme\Desktop\Notebooks & jupyter notebook`
3. Once the shortcut is working as expected, go to Start -> Windows System -> Run
4. Enter `shell:startup` and press enter.
5. This will open the startup folder. Copy the shortcut in this folder. Now whenever the machine starts, the shortcut will be executed.

Saving Task Progress

If a learner partially completed your project (e.g. completing tasks 3 out of 7) and joins the project later to finish the rest, they will get the correct cloud desktop again and they will start from the correct task (task 4). However, the work they did in the old cloud desktop may still be lost.

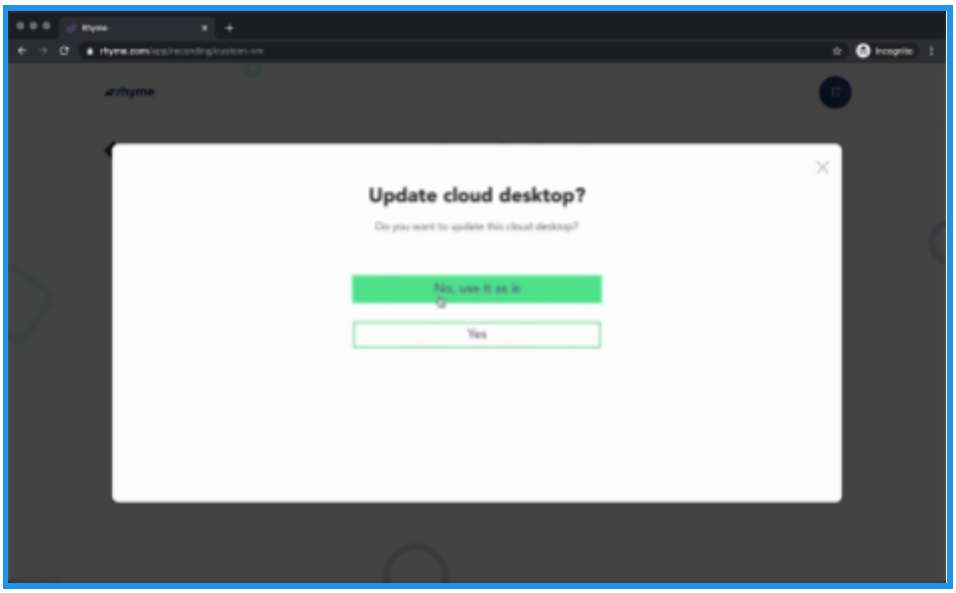
A workaround to that problem can be for you to save a version of your file when you save the task video. For example, if you are working in Excel, save task-1.xlsx, task-2.xlsx, etc.

Then, at the start of the project (i.e., in task 1), tell the learner that if they want to jump to any future task, then they can just open the right numbered file and proceed.

If, on opening such a file, they'll have to run previous cells / cell-blocks again (e.g., in case of tools like Jupyter and RStudio), you can mention that too. You can also consider adding a hint below the applicable cell title. For example, in the case of Jupyter, the hint can be "If you are starting from this task, you can run cells from all previous tasks in the kernel by going to Kernel > Restart and Run All"

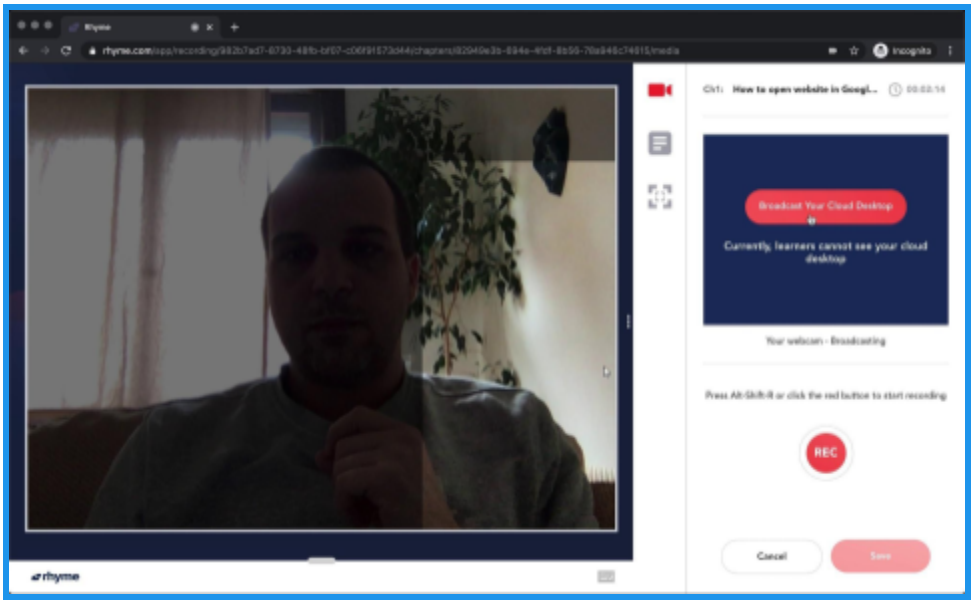
How to Record Videos in Rhyme 💡

How to Create a Recording



Click image to play video

How to Record Video Content



Click image to play video

Video Recording Best Practices

Location

- Choose a quiet place with very little background noise. If you need to listen to audio, use headphones so the speaker audio doesn't echo into the mic.
- Make sure you have a stable internet connection. A wired connection works best.
- Allow Rhyme.com access to your mic. We recommend a wired mic instead of a computer mic. Good audio quality is key to a good recording.
- Turn on your video camera if you plan to show yourself during the introduction. We recommend you do this only at the beginning so the learners can connect and see who you are. Pick a nice background that showcases your location and avoid being in front of a window so you're not backlit.
- Put your phone on silent.
- Quit all unnecessary applications on your computer which may have distracting sound effects or waste CPU/memory. E.g., Slack, Skype, etc.

Welcome Statement

- Start with an overview of "What we are going to do." E.g., In this project, we are going to "learn this" / "experience this" / "make this."
- Start with the "Wow state." Show where we'll be by the end of the project. E.g., By the end of this project, we are going to see awesome results like this.

Post Recording Tips

- After you're done with the recording, preview it to ensure you're happy with it.
- After you're happy with the recording of all tasks, run a test session and attend it as a learner. You'll need to access it as a different user and open another browser or in an incognito window. Check if the learner's copy of the cloud desktop works well, if your instructions are at the right pace, if the instruction video shows the right section of the cloud desktop, etc.

- If you re-record a task, you need to fix your cloud desktop to match the starting point for that task. Also ensure you end the task with the same end state as the starting state of the next task.
- Use annotations to correct small mistakes.
- See Video Editing Features Documentation

Choosing a Session

The session app in Rhyme is the main component that learners interact with. It encapsulates almost all parts of the learners' experience and also the monitoring functionality for guides and TAs. There are three types of sessions.

Session Types

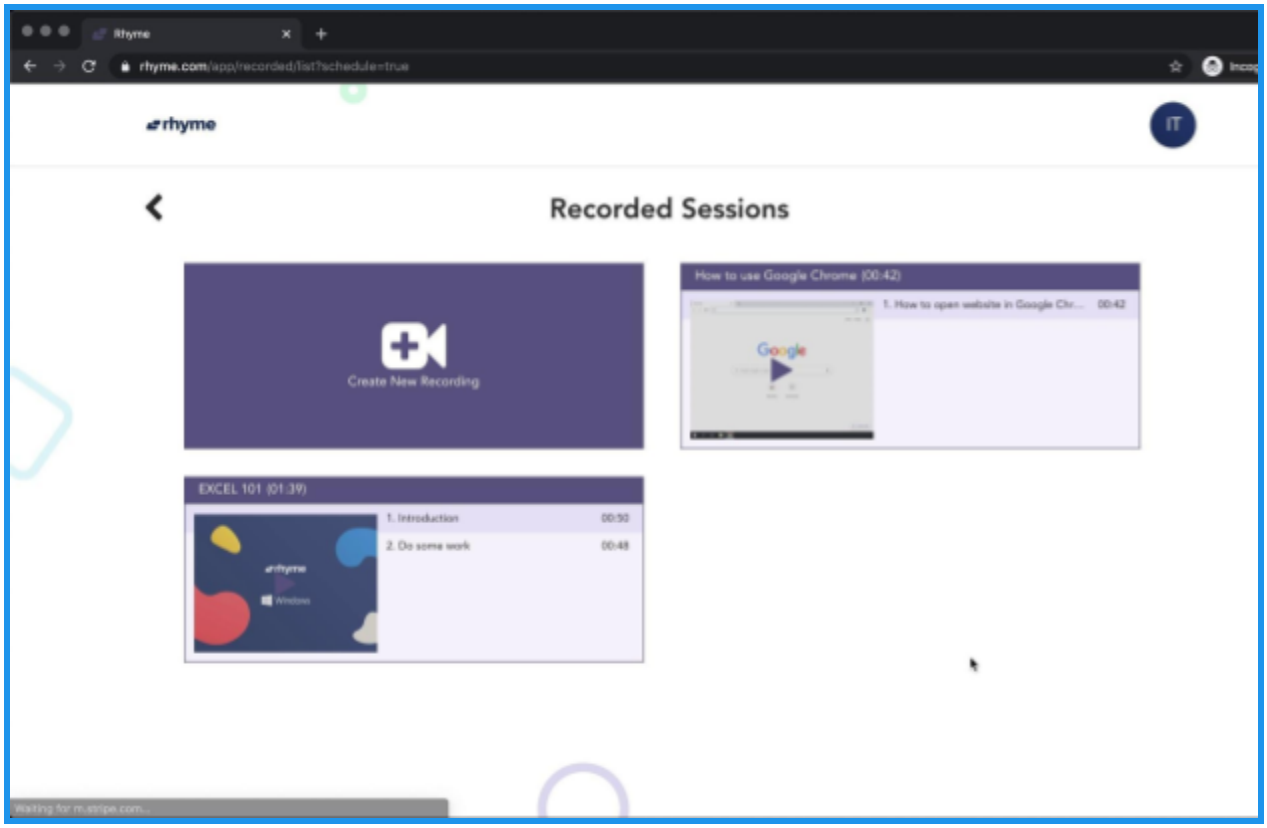
- **Recorded Session** - When we think of projects, we think of recorded sessions. These sessions have a list of tasks recorded for them. Each task includes a video recorded on the same type of cloud desktop that will be later received by the attendees in that session. Those sessions can run with or without a host. If the host is available, he/she can take control of the learner's cloud desktops and help them. Only when control is taken an audio connection is initialized between the guide and the controlled attendee. The recorded session supports up to 2000 learners. The max duration of connection is 10 hours currently, the max session duration is 10 hours via the frontend and 1 year custom. Currently the learner can stay in a recorded session for - sum(project's videos' duration) * 3.
  ⭐ ***This session type is what you will use for your course projects.***

- **Live Session** - Think of live sessions as webinars. They are meant to be scheduled and run only when the instructor/host is available. During these sessions, the host's cloud desktop is broadcasted on the right side. The host has the option to choose some of the learners' Cloud Desktops to be broadcasted. There is an open audio channel between all attendees in the session. The host can take control of attendee cloud desktops. All files uploaded from the host will be uploaded to the cloud desktops of all attendees that are currently online. The live session supports up to 19 learners. The max duration of connection to a cloud desktop is 10 hours. The max duration of the session is 10 hours.
  📖 **Note:** We will not focus on any live sessions for your course projects.

- **Collaboration Mode** - This a specific type of Live Session where one Cloud Desktop is shared between two attendees. This session has a limit of two attendees, where both of them can switch places and control the cloud desktop. There is audio communication enabled between the attendees. The collaboration session supports 2 learners.

  Note: We will not focus on any live sessions for your course projects.

Create a Session



Click image to play video

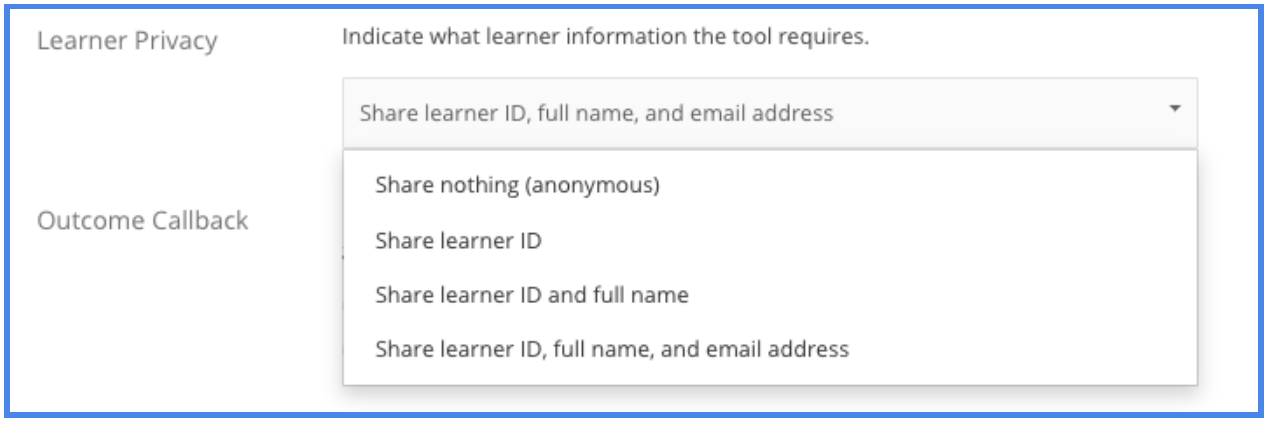
How to Integrate Rhyme to Your Coursera course

Rhyme projects can be embedded into your course content through LTI lesson items. These LTIs will enable learners to launch and authenticate into your Rhyme environment with their Coursera credentials. Once launched to the Rhyme experience, learners can mark their projects as “complete” within Rhyme, which well send a completion indication back to Coursera. Please note that all Rhyme projects are currently ungraded LTI items, so

scores can not be provided for these activities. An accessibility audit for Rhyme features is also currently underway, so please do factor this into your course design process, as you decide whether this tool will fit with your content needs.

Rhyme currently supports LTI 1.1 integration. You'll find the needed configuration for your LTI activity within the Rhyme Session settings for your project recording. From there, you can copy/paste your Launch URL, consumer key, and secret into Coursera LTI settings to complete integration.

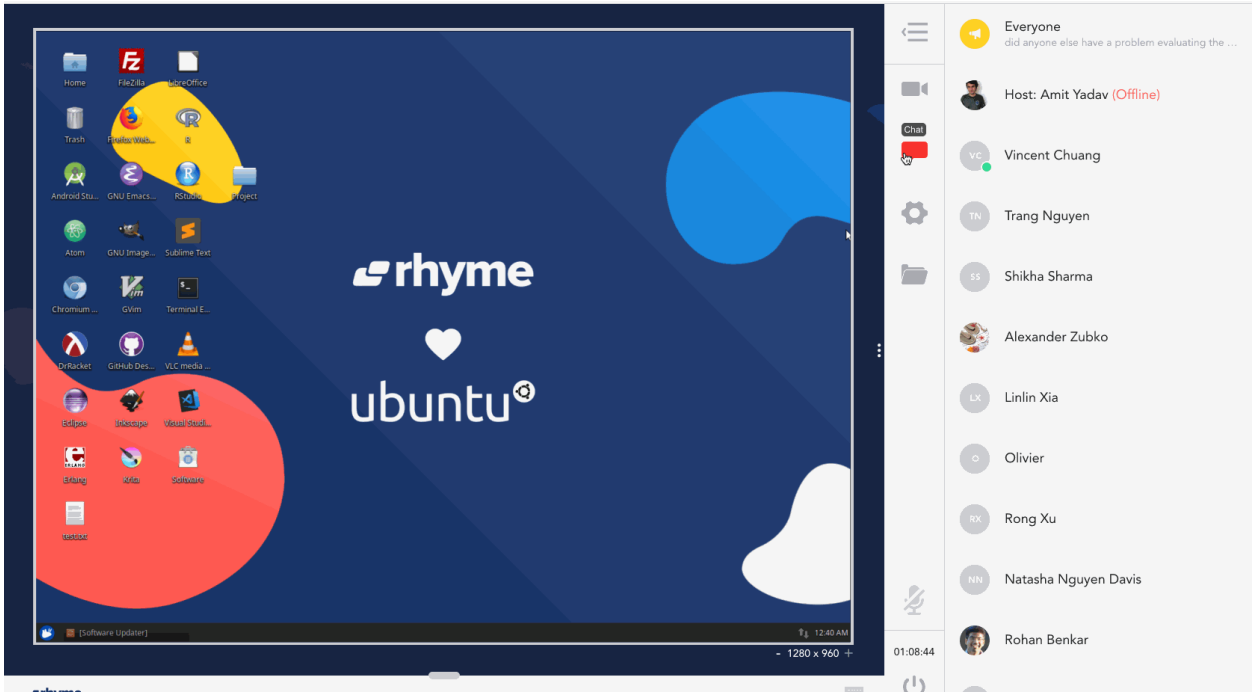
When entering the above fields on Coursera, you will also need to make sure that all the Learner Privacy fields are selected. Please see screenshot below:



Upload & Download Files

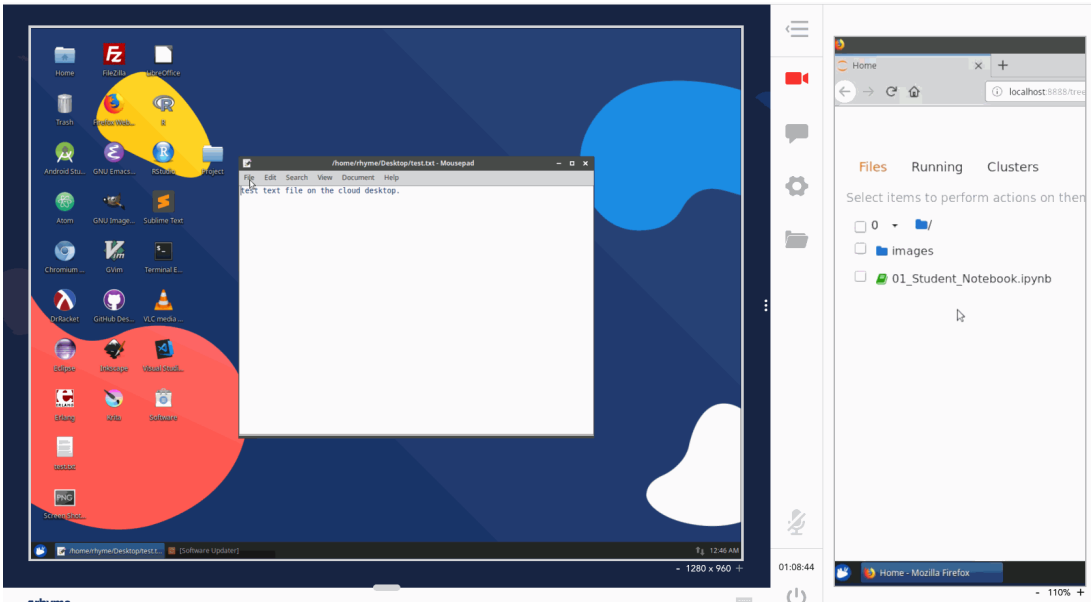
File Upload

Users can easily upload files to their Cloud Desktops. They need to open the “Files” tab from the sidebar, then click on the “Upload files to your cloud desktop” which should prompt the user to select files for upload. If the student is actually a guide in a session, the files will be uploaded to ALL CLOUD DESKTOPS. If the user is just a learner, the files will be uploaded only to his/her Cloud Desktop. If a file upload fails we recommend to the user to use some file sharing service and download the files on the Cloud Desktops after they have been uploaded to https://ufile.io for example.



File Download

Learners may want to save and keep some of the created files from the project. For them to be able to do this, rhyme has a file browser in the same Files tab in the sidebar.
It works similar to normal file browser that lists all the directories and files in a given directory. By default it will start from the “Desktop” of the user. After the user has navigated to the files he/she would like to download, he/she needs to click on the file in the file browser and the file will start downloading.



Instructional Design Questions to Ask When Using Rhyme

- **Are the learning objectives clearly communicated to my learners? Can learners articulate what they will achieve by the end of the project?**
    - Use project description to tell learners the goal of the project. E.g. “By the end of the project, you will be able to …”

- **Does my project need any prerequisites? If yes, do I communicate those with learners? Can learners tell whether they are ready to start the project?**
    - Include prerequisites requirements in the project description and offer a prior knowledge quiz to help learners evaluate their readiness

- **Is my project authentic and reflecting the work in the real world?**
    - Tell learners how this project will help the day-to-day job of a professional. E.g. how pivot tables will help a business analyst in his/her job

- **Is my project too short or too long?**
    - Each project should take less than an hour to complete

- **Do I chunk my project into smaller tasks so that learners can master it step by step?**
    - Each task should take 4-7 mins to complete

- **Do I start my project in an engaging way that stimulates learners' interest and curiosity?**
    - Consider starting the project with a real world challenge or an interesting problem to solve

- **Do I explain each step that I take with clear rationale? Do learners understand why each step is taken?**
    - Explain not only what you do but also ***why*** you do it this way in each step

- **Do I introduce myself at the beginning of the video? Do learners feel I'm approachable and friendly? Do learners feel I'm an expert in my field?**
    - Introduce your name, background, career pathways, and interests in this field at the beginning of the tutorial video

- **Is the screen in my video readable and clearly focused? Am I showing any irrelevant information on the screen that may distract learners' attention?**
    - Make sure the cursor is always where you want learners to focus and the screen is zoomed in a way to cut irrelevant information

- **Can learners complete the project with an end product that they feel proud of and can add to their portfolios?**
    - Show learners how to save their projects after sessions and encourage them to add the projects to their portfolios

- **Can learners transform what they learned from your project to other similar scenarios and solve their own work challenges?**
    - Ask learners to reflect how the skills learned in this project would be helpful for their work. Give learners a similar problem to solve to evaluate their mastery of the skills.