# ATCE (Automated Transfer Credit Evaluator) Tool User Manual

Tyler Dionne Kendall Kelly

# Table of Contents

1. Overview	3
2. Running the Web Application	5
3. Starter Template Website Features	6
3. Steps on How to Modify the Template	10
4. Security Testing Results	11

1. Overview

The ATCE (Automated Transfer Credit Evaluator) Tool is a standalone full stack web application starter template for startup companies looking to have a SaaS model.

The tool comes with a working full stack web application template written using the Flask python web framework.

The website includes the important base features for full stack SaaS applications.

The starter template (atce-webapp.zip) comes with the following resources:

San folders you viewed provinuely	Mar 24, 2025 at 3:48 AM		Folder
See folders you viewed previously	Feb 19, 2025 at 7:17 AM		Folder
📄 арр.ру	Mar 25, 2025 at 2:03 PM	4 KB	Python script
instance	Apr 17, 2025 at 7:10 PM		Folder
📄 site.db	Apr 17, 2025 at 7:10 PM	16 KB	Visualocument
README.md	Jan 23, 2025 at 7:31 PM	100 bytes	Sublimcument
✓	Feb 19, 2025 at 5:20 AM		Folder
✓ i css	Mar 24, 2025 at 3:46 AM		Folder
about.css	Feb 19, 2025 at 4:50 AM	2 KB	Text Document
atce.css	Feb 19, 2025 at 4:53 AM	3 KB	Text Document
🗟 auth.css	Mar 24, 2025 at 4:23 AM	2 KB	Text Document
odocs.css	Feb 19, 2025 at 4:54 AM	1 KB	Text Document
index.css	Feb 19, 2025 at 4:55 AM	2 KB	Text Document
> 🗖 docs	Feb 19, 2025 at 5:20 AM		Folder
∨ 📄 images	Feb 19, 2025 at 5:05 AM		Folder
👕 logo.svg	Jan 23, 2025 at 7:31 PM	14 KB	SVG document
∽ 🛅 js	Feb 19, 2025 at 5:02 AM		Folder
📄 atce.js	Feb 19, 2025 at 5:02 AM	6 KB	JavaScript
🗸 🛅 templates	Mar 24, 2025 at 3:47 AM		Folder
🔹 about.html	Feb 19, 2025 at 6:01 AM	3 KB	HTML text
🔹 atce.html	Feb 19, 2025 at 6:02 AM	3 KB	HTML text
🤞 docs.html	Feb 19, 2025 at 6:00 AM	4 KB	HTML text
index.html	Mar 24, 2025 at 3:56 AM	2 KB	HTML text
s login.html	Mar 24, 2025 at 3:46 AM	3 KB	HTML text
≤ register.html	Mar 24, 2025 at 3:46 AM	3 KB	HTML text
> 🚞 venv	Mar 24, 2025 at 4:16 PM		Folder
📄 virtualenvcomm.txt	Mar 3, 2025 at 12:27 PM	47 bytes	text

The files included in the templates section are the html files for the different website pages.

The app.py file is the main driver for the Flask application and deals with the backend setup with the SQLAIchemy which is currently setup to support user login/registration.

The login/registration setup includes html pages for both login + registration and allows a user to create an account with an email and password. The current implementation performs actions

such as checking to make sure no duplicate accounts are made and all input provided for the forms are error checked.

The files in the static folder deal with the separated styling using css, scripting using javascript and images (logo for business which in this case is the florida tech panther). The atce.js file is the current software/protected tool on the starter template.

#### 2. Running the Web Application

Given that the starter template comes as a standalone full stack Flask application it can be run locally in a few steps.

Simply navigate to the project directory in this case we will use the name "atce-webapp.zip" and run the following command:

\$ python3 app.py

You shall see the following output displayed in the terminal:



Now you may navigate to http://127.0.0.1:5000/ to see the website displayed.



#### 3. Starter Template Website Features

Below demonstrates the current implemented features of the website pages/UI currently.



#### **Our History**

Founded in 1958, Florida Institute of Technology has grown from its origins as "Brevard Engineering College" to become a comprehensive, researchintensive national university with a focus on science, technology, engineering, and mathematics (STEM).

#### **Academic Excellence**

Florida Tech offers more than 200 degree programs at the bachelor's, master's, and doctoral levels. Our rigorous curriculum is designed to challenge students and prepare them for successful careers in rapidly evolving fields.

🎓 NEW: Schedule a virtual campus tour today! 🞓 👘

/docs

# Documentation

### First Semester

Task	Due Date	Documents
Plan	Sep 4	Plan, Presentation
Milestone 1	Sep 30	Requirement, Design, Test, Presentation, Progress Evaluation
Milestone 2	Oct 28	View Presentation, Progress Evaluation
Milestone 3	Nov 25	View Presentation, Progress Evaluation

Second Semester

Task	Due Date	Documents
Plan	San /	View Dian Drecentation

# /login

Log In Username Password I Remember Me Log In	ATCE Tool Catalogs Docs About FIT	Apply Now
Username Password Remember Me Log In Don't have an arrount2 Register	Log In	
Password  Remember Me  Log In  Don't have an arcount? Register	Username	
Remember Me Log in Don't have an account? Register	Password	
Log in	Remember Me	
Don't have an account? Register	Log In	
	Don't have an account? Register	

/register

ATCE Tool Catalogs Docs About FIT	Apply Now
Username	
Email	
Password	
Confirm Password	
Register	
Already have an account? Log In	

# /atce (protected route)

ATCE Tool Catalogs Docs About FIT	Apply Now
Please log in to access this page.	
Username	
Password	
Remember Me	
Log In	
Don't have an account? Register	

ATCE Tool Catalogs Docs About FIT	Apply Now
Automated Transfer Credit Evaluator (ATCE)	
Drag and drop your transcript file here	
or	
Choose File	
Supported formats: PDF, TXT	
© 2024 Florida Institute of Technology	

These images demonstrate the current state of the website along with the currently implemented features that come with the atce-webapp.zip file.

#### 4. Steps on How to Modify the Template

Given that this is the template for a full stack web application the user may want to edit/modify the existing files/structure for their own tool/company.

In order to modify the company image you would swap out the file located at /static/images/logo.svg with the desired company logo.

In order to modify the existing protected software/tool on the website you would swap out the file located at /static/js/atce.js.

In order to modify the webpages you would navigate to /templates and change the html pages based on your company's branding.

In order to change backend configurations/routing and database setup you would navigate to the app.py file.

Through modifying the files in these locations the user shall be able to use the template to create a full stack web application for their company.

# 5. Security Testing Results

The results shown below are the testing results after using Burp Suite to fuzz the login page on the website:

Purpose	Input
Normal test	admin, user1
Long input	a a (1000+ chars)
Special characters	\$\$\$\$\$, <>!@#\$%^&*()
SQL Injection	' OR 1=1, admin'
NoSQL Injection	{"\$ne": null}
XSS Injection	<script>alert(1)</script>
Unicode	ユーザー
Whitespace	\tadmin\t
Path traversal	//etc/passwd
Null byte	admin%00
Quotes/brackets	`"'{0``
Empty	
Command injection	&& /bin/sh\0;

Burp Suite /login Fuzzing Results		
Input #	Input	Result/Output
01 (User + Pass)	admin	Normal (Login unsuccessful. Please check username and password.)
02 (User + Pass)	aaa aaa (1000 a's)	Normal (Login unsuccessful. Please check username and password.)
03 (User + Pass)	\$\$\$\$\$	Normal (Login unsuccessful. Please check username and password.)
04 (User + Pass)	<>!@#\$%^&*()	Normal (Login unsuccessful. Please check username and password.)

05 (User + Pass)	' OR 1=1, admin'	Normal (Login unsuccessful. Please check username and password.)
06 (User + Pass)	{"\$ne": null}	Normal (Login unsuccessful. Please check username and password.)
07 (User + Pass)	<script>alert(1)</script>	Normal (Login unsuccessful. Please check username and password.)
08 (User + Pass)	ユーザー	Normal (Login unsuccessful. Please check username and password.)
09 (User + Pass)	\tadmin\t	Normal (Login unsuccessful. Please check username and password.)
10 (User + Pass)	//etc/passwd	Normal (Login unsuccessful. Please check username and password.)
11 (User + Pass)	admin%00	Normal (Login unsuccessful. Please check username and password.)
12 (User + Pass)	((3)	"Please fill out this field." displayed properly
13 (User + Pass)	&& /bin/sh\0;	Normal (Login unsuccessful. Please check username and password.)

The image shown below displays the results upon testing the web application with OWASP ZAP's automated scan:

