The module and the functional testing:

In the "login" module. The function for logging into an account.

     Inputs: email:string, password:string

Forming equivalence classes:

Equ. Class	Description	Possible Values
$EC_{E,valid}$	Email that exists in the database	abc@gmail.com
$EC_{E,invalid}$	Email that DNE	abc@, @gmail.com, abc
$EC_{E,empty}$	Email is empty	" "
$EC_{P,valid}$	Password that exists in the database	ASCII characters
$EC_{P,invalid}$	Password that DNE in the database	ASCII characters
$EC_{P,empty}$	Password is empty	" "

Derive test cases (for valid inputs):

Equ. Class	Input		Expected Output
	Email	**Password**	
$EC_{E,valid}$ and $EC_{P,valid}$	joe@gmail.com	pass123	Redirect to home page

Derive test cases (for invalid inputs):

Equ. Class	Input		Expected Output
	Email	**Password**	
$EC_{E,valid}$ and $EC_{P,invalid}$	abc@gmail.com	p	Redirect to login page
$EC_{E,valid}$ and $EC_{P,empty}$	abc@gmail.com	" "	Redirect to login page
$EC_{E,invalid}$ and $EC_{P,valid}$	abc	password	Please include an '@' in the email address. 'abc' is missing an '@'.
$EC_{E,invalid}$ and $EC_{P,invalid}$	abc@	pa	Please enter a part following '@'. 'abc@' is incomplete.
$EC_{E,invalid}$ and $EC_{P,empty}$	def@gmail.com	" "	Redirect to login page

$EC_{E,empty}$ and $EC_{P,valid}$	" "	password	Redirect to login page
$EC_{E,empty}$ and $EC_{P,invalid}$	" "	pas	
$EC_{E,empty}$ and $EC_{P,empty}$	" "	" "	Redirect to login page