**Summary: Struct IssueView: View**

* Editing view of the issue
* The From allows for the the user to interactive with the issues.
* Edit the title of the issue select a priority and describe the issue

**Struct IssueView: View**

Variables

@ObservedObject var issue: Issue

* Watches the issue for any updates

@EnviromentObject var dataController: DataController

Views

Body:

Form

* To build the user interface to allow
* Two different sections
  + Section 1
    - Edit/enter the issue title
    - displays the date
    - Select the picker priority of he issue
    - Menu to select multiple tags and add them to a specific issue
  + Section 2
    - User enters the description of the issue and this updates issueContent in the data model
* Disable deleting if the issue is deleted
  + so users can delete an issue in landscape mode and return to try and continue editing that issue because memory will be lost
* onRecieve(issue.objectWillChange)
  + watch for any changes to issue then call dataController.queueSave()
* Picker
  + Connected to the priority in core data to user choice of priority
  + Priority
    - Choosing low/medium/high
    - added a tag and converted the tag to int16
* Description of the problem
  + VStack with text basic information
  + Textfield for the User to write description

* Menu
  + Filters through the selected and unselected tags
  + Each issue can add more tags
  + Shows the selected tags with a checkmark
  + Divides with unselected tags

Functions