Model Development Phase Template

Date	18 June 2025
Team ID	SWTID1749622322
Project Title	HealthCareApp – Mental Health Prediction Model Using ML
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

Decision tree model:

```
dtc = DecisionTreeClassifier()
dtc.fit(X_train, y_train)
y_pred = dtc.predict(X_test)

dtc_train_acc = accuracy_score(y_train, dtc.predict(X_train))
dtc_test_acc = accuracy_score(y_test, y_pred)
```

Random forest model:

```
rfc = RandomForestClassifier(
    criterion='entropy',
    max_depth=10,
    max_features='sqrt',
    min_samples_leaf=1,
    min_samples_split=3,
    n_estimators=140,
    random_state=42
)

rfc.fit(X_train, y_train)

y_pred = rfc.predict(X_test)

rfc_train_acc = accuracy_score(y_train, rfc.predict(X_train))
rfc_test_acc = accuracy_score(y_test, y_pred)
```

KNN model:

```
knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train, y_train)
y_pred = knn.predict(X_test)

knn_train_acc = accuracy_score(y_train, knn.predict(X_train))
knn_test_acc = accuracy_score(y_test, y_pred)
```

Logistic Regression model:

```
lr = LogisticRegression(max_iter=1000)
lr.fit(X_train, y_train)
y_pred = lr.predict(X_test)

lr_train_acc = accuracy_score(y_train, lr.predict(X_train))
lr_test_acc = accuracy_score(y_test, y_pred)
```

Gaussian Naïve Bayes:

```
nb = GaussianNB()
nb.fit(X_train, y_train)
y_pred = nb.predict(X_test)

nb_train_acc = accuracy_score(y_train, nb.predict(X_train))
nb_test_acc = accuracy_score(y_test, y_pred)
```

SVC model:

```
svc = SVC(kernel='rbf')
svc.fit(X_train, y_train)
y_pred = svc.predict(X_test)

svc_train_acc = accuracy_score(y_train, svc.predict(X_train))
svc_test_acc = accuracy_score(y_test, y_pred)
```

ADA Booster Classifer:

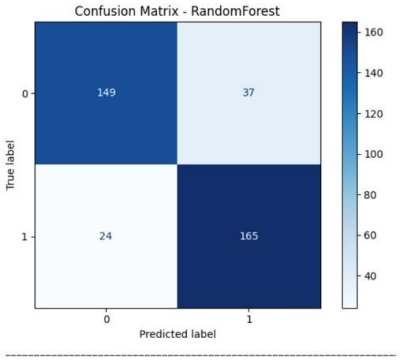
```
abc = AdaBoostClassifier(n_estimators=100, learning_rate=1.0)
abc.fit(X_train, y_train)
y_pred = abc.predict(X_test)

abc_train_acc = accuracy_score(y_train, abc.predict(X_train))
abc_test_acc = accuracy_score(y_test, y_pred)
```

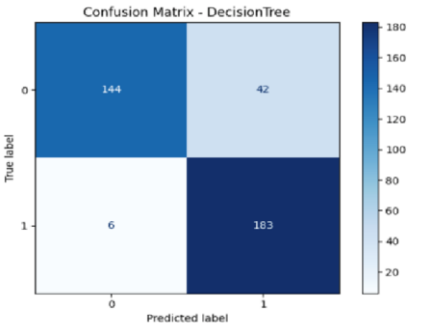
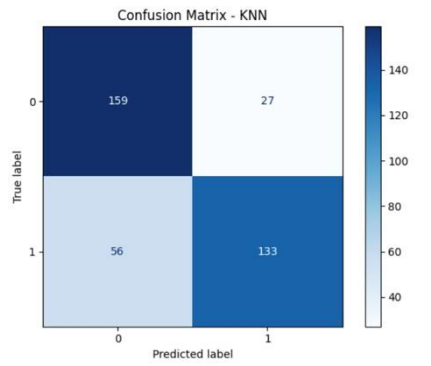
Gradient Boosting Classifier:

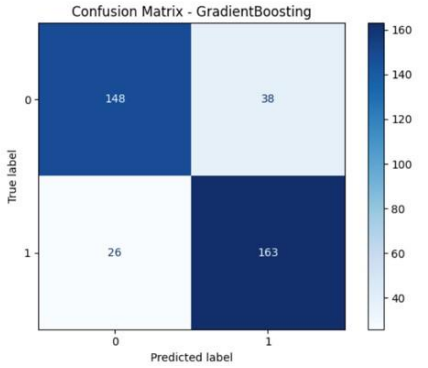
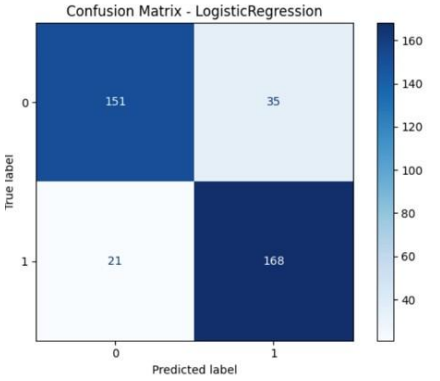
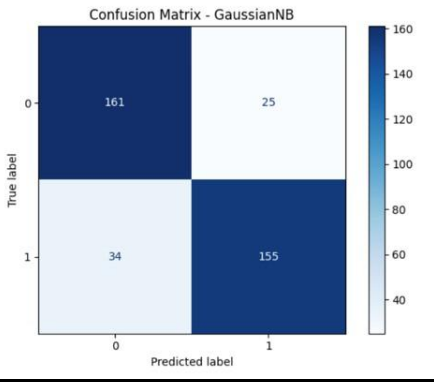
```
abc = AdaBoostClassifier(n_estimators=100, learning_rate=1.0)
abc.fit(X_train, y_train)
y_pred = abc.predict(X_test)

abc_train_acc = accuracy_score(y_train, abc.predict(X_train))
abc_test_acc = accuracy_score(y_test, y_pred)
```

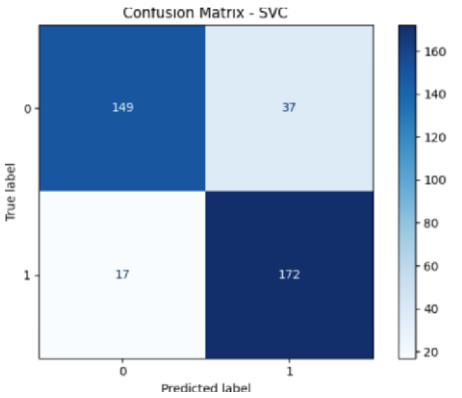
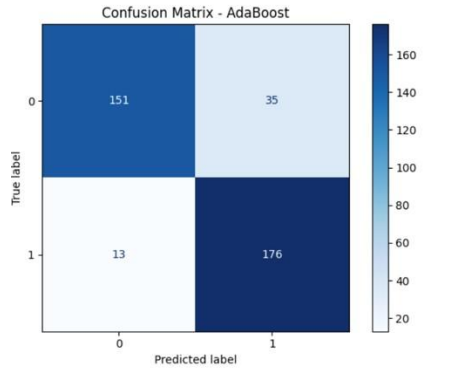
XGBoost Classifier:

```
xgb = XGBClassifier(n_estimators=100, learning_rate=0.1, max_depth=3, use_label_encoder=False, eval_metric='logl
xgb.fit(X_train, y_train)
y_pred = xgb.predict(X_test)

xgb_train_acc = accuracy_score(y_train, xgb.predict(X_train))
xgb_test_acc = accuracy_score(y_test, y_pred)
```

Model	Classification Report	F1 Score	Confusion Matrix
Random Forest	Classification Report for RandomForest: precision recall f1-score support 0 0.86 0.80 0.83 186 1 0.82 0.87 0.84 189 accuracy 0.84 375 macro avg 0.84 0.84 0.84 375 weighted avg 0.84 0.84 0.84 375	83.73%	

Decision Tree	Classification Report for DecisionTree: precision recall f1-score support 0 0.96 0.77 0.86 186 1 0.81 0.97 0.88 189 accuracy 0.87 375 macro avg 0.89 0.87 0.87 375 weighted avg 0.89 0.87 0.87 375	76.27%	
KNN	Classification Report for KNN: precision recall f1-score support 0 0.74 0.85 0.79 186 1 0.83 0.70 0.76 189 accuracy 0.78 375 macro avg 0.79 0.78 0.78 375 weighted avg 0.79 0.78 0.78 375	77.33%	

Gradient Boosting	Classification Report for GradientBoosting: precision recall f1-score support 0 0.85 0.80 0.82 186 1 0.81 0.86 0.84 189 accuracy 0.83 375 macro avg 0.83 0.83 0.83 375 weighted avg 0.83 0.83 0.83 375	84%	
Logistic Regression	Classification Report for LogisticRegression: precision recall f1-score support 0 0.88 0.81 0.84 186 1 0.83 0.89 0.86 189 accuracy 0.85 375 macro avg 0.85 0.85 0.85 375 weighted avg 0.85 0.85 0.85 375	85.07%	
GaussianNb	Classification Report for GaussianNB: precision recall f1-score support 0 0.83 0.87 0.85 186 1 0.86 0.82 0.84 189 accuracy 0.84 375 macro avg 0.84 0.84 0.84 375 weighted avg 0.84 0.84 0.84 375	84.53%	

SVC model	Classification Report for SVC: precision recall f1-score support 0 0.90 0.80 0.85 186 1 0.82 0.91 0.86 189 accuracy 0.86 375 macro avg 0.86 0.86 0.86 375 weighted avg 0.86 0.86 0.86 375	85.33%	
AdaBoost	Classification Report for AdaBoost: precision recall f1-score support 0 0.92 0.81 0.86 186 1 0.83 0.93 0.88 189 accuracy 0.87 375 macro avg 0.88 0.87 0.87 375 weighted avg 0.88 0.87 0.87 375	86.93%	
XG Boost	Classification Report for XGBoost: precision recall f1-score support 0 0.87 0.82 0.85 186 1 0.83 0.88 0.86 189 accuracy 0.85 375 macro avg 0.85 0.85 0.85 375 weighted avg 0.85 0.85 0.85 375	84%